MICROHOBBY SEMANAL

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

AÑO 1 - N.º 5

95 PTAS. Canarias 105 ptas.

HOP EDITA HOBBY PRESS S.A.

HARDWARE

AMPLIA LA MEMORIA DE TU SPECTRUM A 64 K.

GRAN CONCURSO

DUELO ENTRE TITANES

PROGRAMAS

- ASALTO AL CASTILLO
- DISEÑA TUS GRAFICOS EN TRES DIMENSIONES

¡REGALAMOS UN 'QL' CADA MES!

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo
Domingo Gómez

Redactor Jefe
Africa Pérez Tolosa

Diseño
Jesús Iniesta

Maqueta
Rosa M. Capitel

Redacción
José María Díaz
Gabriel Nieto

Colaboradores
Jesús Alonso, Lorenzo Cebeira, Primitivo de Francisco, Rafael Prades, Víctor Prieto

Fotografía
Javier Martínez

Dibujos
José M. Ponce, Fernando Hoyos, Manuel Berrocal, J.M. Ballesteros

Edita
HOBBY PRESS, S.A.

Presidente
María Andrino

Consejero Delegado
José I. Gómez-Centurión

Administrador General
Ernesto Marco

Jefe de Publicidad
Marisa Esteban

Secretaria de Publicidad
Concha Gutiérrez

Publicidad Barcelona
Isidro Iglesias
Tel.: (93) 307 11 13

Secretaria de Dirección
Marisa Cogorro

Suscripciones
M.ª Rosa González
M.ª del Mar Calzada

Redacción, Administración y Publicidad
Arzobispo Morcillo, 24, oficina 4.
28029 Madrid
Telf.: 733 50 12

Distribución
Coedis, S.A. Valencia, 245.
Barcelona.

Imprime
Rotedic, S.A.

Fotocomposición
Consulgraf
Nicolás Morales, 34 - 1.º
Tel.: 471 29 08

Fotomecánica
Zescán
Nicolás Morales, 38
Tel.: 472 38 58

Depósito Legal:
M-36.598-1984

Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América, 1.532. Telf.: 21 24 64. 1209 BUENOS AIRES (Argentina).

Derechos Exclusivos
«Sinclair Users», «Sinclair Programs» y «Sinclair Projects» de EMAP Publications (Londres).

Se solicitará control OJD

MICROHOBBY ESTA SEMANA

Año I · N.º 5 · 4 al 10 de diciembre de 1984
95 ptas. (Sobretasa Canarias 10 ptas.)



ROTULOS MOVILES



Cada vez que decidimos adornar uno de nuestros programas con textos explicativos o instrucciones de uso, nos encontramos con el mismo problema: resulta excesivamente escueto y no aclara nada o es tan largo que su presentación en pantalla nos estropea parte de la zona que habíamos destinado a los gráficos, etc.

Además de todo esto, cada vez que se imprime un texto, es obligado borrarlo una vez transcurrido un tiempo prudencial para su lectura.

La solución adoptada en este caso es bastante simple y mejora notablemente la estética del rótulo. Consiste en una presentación en «ventana». Es decir, no muestra todo el texto a la vez, sino sucesivamente.

Se trata de unas cuantas instrucciones que han sido pensadas para incorporarse como subrutina dentro de su propio programa, en cuyo caso es necesario hacer un simple cambio: sustituir a$ por el texto adecuado en cada caso, y añadir al final un RETURN.

Todas las variables de la línea 2Ø pueden ser modificadas para adaptarse a sus propias necesidades. La «x» y la «y» indican las coordenadas del inicio del texto, mientras que la «z» marca la longitud de la «ventana» de presentación del mismo.

Por último, la PAUSE de la línea 9Ø puede ser modificada para acelerar o retardar el efecto.

```
  10 LET a$="Prueba para ver com
o funciona esta rutina"
  20 LET x=12: LET y=8: LET z=12
  30 DIM b$(25)
  40 LET a$=b$+a$
  50 FOR n=1 TO LEN a$
  60 LET a$=a$(2 TO )+CHR$ 32
  70 PRINT AT x,y; INK 7; PAPER
3;a$(z TO 22)
  80 BEEP .01,20
  90 PAUSE 5
 100 NEXT n
```

TEXTOS EN VERTICAL

Bastante similar al truco anterior, éste nos permite imprimir textos verticales en cualquier posición de la pantalla, con la única limitación de no tener más de 2Ø caracteres de longitud (de lo contrario, aparecería el mensaje «out of screen»).

También puede usarse como subrutina añadiendo un RETURN al final y sustituyendo a$ por el texto pertinente.

Las variables «x» y «y» marcan las coordenadas de origen.

Se ha previsto también una pequeña rutina de borrado a la que se tiene acceso pulsando cualquier tecla, una vez impreso el texto. Si se incorpora como subrutina, ha de sustituirse la línea 12Ø por un PAUSE que marque el intervalo entre la impresión del texto y su correspondiente borrado.

```
  10 LET a$="* Scroll * Vertical
 *"
  20 LET x=0: LET y=0
  30 FOR n=1 TO LEN a$
  40 BEEP .01,20
  50 PAUSE 5
  60 FOR n=1 TO 21
  70 PRINT AT x+n,y; INK 7; PAPE
R 3;a$(n)
  80 BEEP .01,n*2
  90 PAUSE 5
 100 NEXT n
 120 PRINT #1;"      PULSE CUALQU
IER TECLA": PAUSE 0
 130 FOR n=1 TO 21
 140 PRINT AT x+n,y; PAPER 7;CHR
$ 32
 150 BEEP .01,40-n*2
 160 NEXT n
```

¿MAYUSCULAS O MINUSCULAS?

Es muy corriente que en un programa de los denominados interactivos, sea necesario mantener un diálogo con el usuario y controlar la validez de su respuesta.

Una vez más, existen muchos métodos de llevar esto a cabo y muchos tipos de «respuestas». Nosotros nos centraremos en el caso en el que se exija que la letra o letras de la respuesta sea mayúscula o minúscula.

Existe una variable del sistema que controla el modo del cursor, situada en la dirección 23658 decimal.

Esta posición de memoria almacena Ø para cursor en modo «L» (minúsculas) y 8 en modo «C» (mayúsculas); así que bastará colocar en el lugar adecuado POKE 23658, Ø o POKE 23658, 8 para ahorrarnos sentencias del tipo: IF INKEY$ = "N" OR INKEY$ = "n" THEN...

VELOCIDAD DE AUTORREPETICION DE LAS TECLAS

Cuando tenemos que teclear un programa muy largo, o introducir una enorme y tediosa serie de DATAS, o borrar largas líneas, puede resultarnos de utilidad controlar la velocidad de autorrepetición de las teclas.

Para ello (¡cómo no!), existen dos variables del sistema, REPDEL y REPPER, que gobiernan el tiempo que debe transcurrir para que la tecla se repita y el intervalo entre sucesivas repeticiones respectivamente.

Los valores iniciales son REPDEL, 35, REPPER, 5.

Estas variables se encuentran localizadas en las posiciones 23651 y 23652. Con poke 23651 (2), valor puede controlarse perfectamente la velocidad de las teclas, por ejemplo: POKE 23652,255 lo hará lentísimo.

¡¡GRAN CONCURSO MASTER-MIND!!

ORGANIZADO POR MICRO WORLD Y MICROHOBBY SEMANAL

Este gran concurso, que por primera vez en España va a enfrentar en competición a programas de ordenador, le brinda la oportunidad de demostrar que es el mejor programando. Y por supuesto, de llevarse grandes premios.
En el concurso participarán programas que jueguen al «Mastermind» (en una modalidad determinada que se explica más adelante), y la forma de seleccionar y elegir al mejor es la competición entre todos. De esta forma, el programa ganador habrá demostrado que es el mejor, al haber superado y eliminado a todos sus contrincantes.

DESCRIPCION GENERAL DEL JUEGO

El juego consiste en que cada programa debe intentar acertar una secuencia de números aleatoria y secreta generada por el otro programa, antes de que el otro programa acierte la secuencia generada por él.

Para intentar conseguirlo, cada uno de los programas irá proponiendo secuencias de números basadas en las «pistas» que el otro programa le vaya dando.

Estas «pistas» estarán referidas a los números que de cada secuencia se vayan acertando, así como a la posición que ocupen dentro de la misma.

Convendremos en que a los aciertos plenos (número y posición), les llamaremos «muertos» (M) y a los aciertos de números sin la posición correcta, les llamaremos «heridos» (H).

Así, si un programa ha conseguido adivinar dos de los números de la secuencia generada por el otro, éste responderá «2H». Pero si uno de ellos está en la posición correcta, entonces deberá responder «1H 1M».

Ganará el programa que consiga acertar primero la secuencia secreta generada por el otro.

En caso de que el programa que empezó primero, acierte ante la secuencia generada por el otro, se le dará a éste una última oportunidad de conseguirlo. Si lo logra, se llegaría a un empate en el juego. Los empates en cada juego, se resolverán mediante un nuevo juego.

REGLAS DEL JUEGO (PROGRAMA)

El programa debe generar una secuencia aleatoria de cinco números, comprendidos entre el 1 y el 9. En esta secuencia no debe haber repeticiones de números, y será secreta para el otro ordenador, pero deberá aparecer en pantalla con el siguiente mensaje:

SECUENCIA GENERADA: nnnnn

A continuación, cada programa debe preguntar quién empieza a jugar primero, con el siguiente mensaje:

COMIENZO YO A JUGAR (S/N): ?

y quedará a la espera de recibir la respuesta, que evidentemente sólo podrá ser una "S" o una "N".

El programa que empiece primero, propondrá una secuencia numérica aleatoria y esperará a que se le introduzca la pista (respuesta) facilitada por el otro programa, así como también la secuencia propuesta por otro programa.

El programa que empezó en segundo lugar, quedará a la espera de recibir la secuencia propuesta por el primero, a la que deberá responder con su pista (respuesta) y su secuencia propuesta, quedando de nuevo a la espera de recibir la pista (respuesta) y la secuencia del que empezó primero.

Este ciclo deberá repetirse hasta que uno de los dos acierte plenamente la secuencia secreta generada por el otro.

El tiempo máximo de respuesta de cada jugada no puede ser superior a 4 minutos.

FORMA DE SELECCION

Los programas admitidos al consurso entrarán en la primera fase del mismo. En esta primera fase se hará competir a los programas en grupos de dos, cargando cada uno de ellos en un ordenador Sinclair ZX Spectrum de 48 K, y jugando una partida. Además cada una de ellas se jugará a dos juegos, comenzando cada vez uno de los dos programas. El programa que pierda los dos juegos quedará eliminado, pasando a la segunda fase el programa que ha ganado los dos. En caso de empate, esto es, si cada uno gana un juego, pasarán ambos a la segunda fase.

En cada partida habrá un operador-árbitro que introduce las jugadas de cada ordenador en el otro. Evidentemente, este operador-árbitro no influye en el juego. Si el autor del programa concursante asiste a la partida, le estará permitido a él mismo introducir la respuesta del otro ordenador en su programa, siempre en presencia del operador-árbitro.



BASES

1. Todos los programas que se presenten deberán «correr» sobre un ordenador Sinclair ZX Spectrum de 48 K.
2. Todos los programas deberán ser originales.
3. Cualquier programa que durante su ejecución, se interrumpa presentando mensajes de error, será automáticamente descalificado.
4. Todos los programas deberán ajustarse a las reglas de juego que aquí se detallan.
5. Los programas deberán enviarse grabados en cassette, con el original por una cara y una copia por la otra.
6. Todas las partidas serán públicas, pudiendo asistir a ellas cuantas personas lo deséen.
7. Tanto el calendario con las partidas a celebrarse como la fecha, lugar y hora de las mismas, se publicarán con la suficiente antelación y siempre, desde las páginas de esta revista.
8. La participación en el concurso supone la aceptación de estas bases, por lo que quedarán automáticamente eliminados aquellos programas que no se ajusten estrictamente a las mismas.
9. No podrán presentarse a este concurso ningún empleado ni familiar de la editorial Hobby Press, ni de la firma Microworld.

PREMIOS

Un capítulo importante de este gran concurso es el de los premios que recibirán los diez primeros finalistas.

En este sentido se distribuirán de la siguiente manera:

- Primer premio: un viaje a Londres para dos personas.
- Segundo premio: un monitor de color.
- Tercer premio: un Spectrum Plus.
- Cuarto, quinto, sexto, septimo, octavo, noveno y décimo: una serie de lotes de programas de Microparadise y Dinamic.

¡PARTICIPE Y SUERTE!

CONTOUR

Spectrum 16 K.

Este fascinante programa permite crear sorprendentes formas tridimensionales de la manera más simple del mundo. Emplee a fondo su imaginación y dispóngase a dibujar. Además de esto, aprovechamos para llamarle la atención sobre la sencillez del algoritmo utilizado.

La pantalla del ordenador se convierte en una pizarra dividida en cuatro cuadrantes unidos de dos en dos, como se verá con mayor claridad al utilizar el programa.

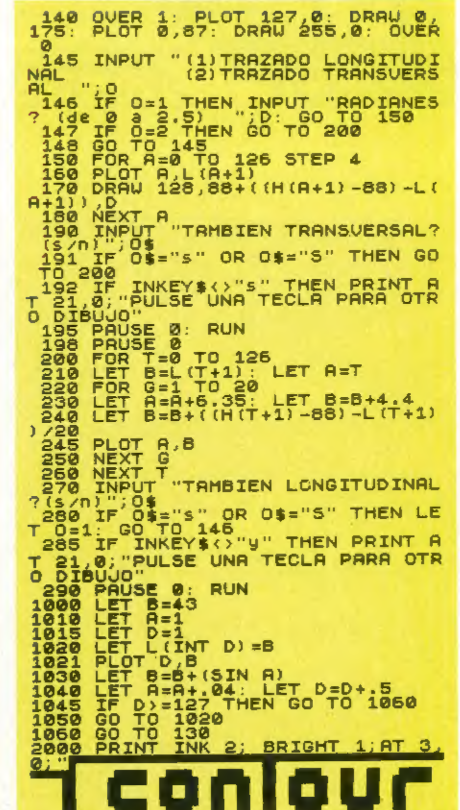
Empleando las teclas del 5 al 9, podemos dibujar líneas con distinta inclinación; concretamente:

—(5) traza hacia abajo con una inclinación de 45 grados.
—(6) hacia abajo. Inclinación de 30 grados.
—(7) hacia arriba. Inclinación de 30 grados.
—(8) hacia arriba. Inclinación de 45 grados.
—(9) traza línea horizontal.

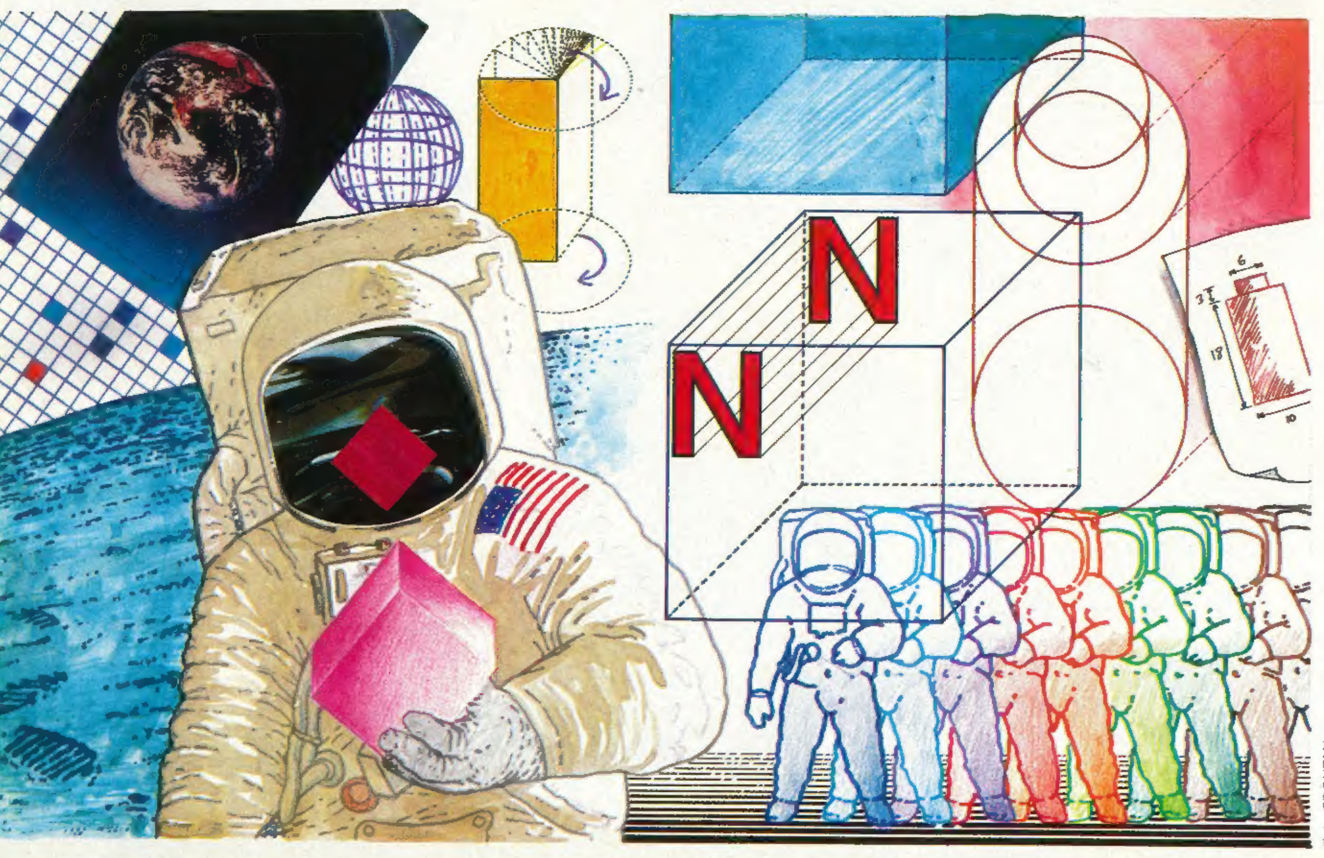
Existe además una tecla especial, el «∅», con el cual dibujamos una onda sinusoidal, siempre de las mismas características; aunque esto puede parecer algo excéntrico, la verdad es que es muy útil para realizar dibujos en los que necesitemos estructuras con muchos pliegues.

Una vez que hemos completado el esqueleto del dibujo, el programa se encarga de rellenarlo preguntándonos si queremos un trazado longitudinal o transversal y la inclinación del mismo.

```
   5 REM
   6 GO SUB 2000
  10 PLOT 127,0: DRAW 0,175: PLO
T 0,87: DRAW 255,0
  20 DIM L(127): DIM H(127)
  30 LET A=43
  35 LET D=-1
  40 FOR T=1 TO 2
  50 FOR G=1 TO 127
  60 IF INKEY$="9" THEN PLOT G+D
,A: GO TO 100
  70 IF INKEY$="8" THEN LET A=A+
1: PLOT G+D,A: GO TO 100
  75 IF INKEY$="5" THEN LET A=A-
1: PLOT G+D,A: GO TO 100
  80 IF INKEY$="7" THEN LET A=A+
.5: PLOT G+D,A: GO TO 100
  90 IF INKEY$="6" THEN LET A=A-
.5: PLOT G+D,A: GO TO 100
  91 IF INKEY$="0" THEN GO TO 10
00
  95 GO TO 60
 100 IF T=1 THEN LET L(G)=A: GO
TO 115
 110 LET H(G)=A
 115 NEXT G
 130 LET A=131: LET D=127: NEXT
T
```



```
 140 OVER 1: PLOT 127,0: DRAW 0,
175: PLOT 0,87: DRAW 255,0: OVER
 0
 145 INPUT "(1)TRAZADO LONGITUDI
NAL          (2)TRAZADO TRANSVERS
AL  ";O
 146 IF O=1 THEN INPUT "RADIANES
? (de 0 a 2.5)   ";D: GO TO 150
 147 IF O=2 THEN GO TO 200
 148 GO TO 145
 150 FOR A=0 TO 126 STEP 4
 160 PLOT A,L(A+1)
 170 DRAW 128,88+((H(A+1)-88)-L(
A+1)),D
 180 NEXT A
 190 INPUT "TAMBIEN TRANSVERSAL?
(s/n)";O$
 191 IF O$="s" OR O$="S" THEN GO
 TO 200
 192 IF INKEY$<>"s" THEN PRINT A
T 21,0;"PULSE UNA TECLA PARA OTR
O DIBUJO"
 195 PAUSE 0: RUN
 198 PAUSE 0
 200 FOR T=0 TO 126
 210 LET B=L(T+1): LET A=T
 220 FOR G=1 TO 20
 230 LET A=A+6.35: LET B=B+4.4
 240 LET B=B+((H(T+1)-88)-L(T+1)
)/20
 245 PLOT A,B
 250 NEXT G
 260 NEXT T
 270 INPUT "TAMBIEN LONGITUDINAL
?(s/n)";O$
 280 IF O$="s" OR O$="S" THEN LE
T O=1: GO TO 146
 285 IF INKEY$<>"y" THEN PRINT A
T 21,0;"PULSE UNA TECLA PARA OTR
O DIBUJO"
 290 PAUSE 0: RUN
1000 LET B=43
1010 LET A=1
1015 LET D=1
1020 LET L(INT D)=B
1021 PLOT D,B
1030 LET B=B+(SIN A)
1040 LET A=A+.04: LET D=D+.5
1045 IF D>=127 THEN GO TO 1060
1050 GO TO 1020
1060 GO TO 130
2000 PRINT INK 2; BRIGHT 1;AT 3,
0;"contour"
2010 PRINT AT 10,15;"cursores";A
T 10,15; OVER 1;"_________"
2020 PRINT AT 12,10;"(5) Baja a
45 grados"
2030 PRINT AT 13,10;"(6) Baja a
30 grados"
2040 PRINT AT 14,10;"(7) Sube a
30 grados"
2050 PRINT AT 15,10;"(8) Sube a
45 grados"
2060 PRINT AT 16,10;"(9) Horizon
tal"
2070 PRINT AT 17,10;"(0) Ondas"
2080 PRINT AT 21,0;"PULSE UNA TE
CLA PARA COMENZAR..."
2090 PAUSE 0: CLS : RETURN
```



F. L. FRONTAN

Con este programa no se sentirá en el espacio, pero logrará las formas más sorprendentes.

ASALTO AL CASTILLO

Spectrum 16 K

Nos encontramos en esta ocasión con un programa divertido que nos obligará a mantener nuestra atención muy fija en la pantalla y nos hará recordar grandes hazañas de las que nos sentiremos protagonistas.

Si se acuerda usted de las emocionantes películas del cine y la televisión en las que una furiosa multitud se dirigía al asalto de un castillo medieval, pues felicidades; mediante la magia de los bits, se convertirá en el artífice de su defensa.

Mientras sus enemigos ascienden decididamente por la muralla, usted controla una enorme piedra que puede destruirles si les toca; hasta aquí parece fácil, pero los «escaladores» son muchos y sólo posee una bola. Si alcanza los 1.500 puntos, puede considerarse un auténtico Napoleón.

Los controles del juego son muy sencillos:

—«p» mueve la bola hacia la derecha.
—«o» hacia la izquierda.
—«m» ¡fuego!

Dos advertencias: las teclas deben estar en modo minúsculas y la respuesta debe ser rápida.



M. BERROCAL

Como en el dibujo, intente defender el castillo contra el ataque de los intrépidos asaltantes.

NOTAS GRAFICAS

B D E F

```
   5 GO SUB 9600
   8 BORDER 1: PAPER 5: BRIGHT 1
  10 PRINT INK 2;AT 3,9;"▀▄▀▄▀▄▀▄"
  15 PRINT AT 4,9; BRIGHT 1; FLA
SH 1;"COBBER  CASTLE"
  20 PRINT INK 2;AT 5,9;"████████"
  30 PRINT AT 12,9;"Izquierda   =
 O"
  35 PRINT AT 14,9;"Derecha     =
 P"
  40 PRINT AT 16,9;"Tirar bolas=
 M"
  50 PRINT AT 21,0;" PULSE UNA T
ECLA PARA COMENZAR "
  55 PAUSE 0
  60 BORDER 5: BRIGHT 1: PAPER 5
: CLS
  70 DIM b(10,4)
  80 LET a=5: LET sc=0
  90 PRINT AT 2,2;"COBBER CASTLE
"
  95 PRINT AT 2,20;"PUNTOS=  0"
 100 PRINT AT 4,31;""
 110 FOR f=1 TO 17*32
 120 PRINT PAPER 6; INK 2;"╤";
 130 NEXT f
 140 GO SUB 7000
 200 FOR f=1 TO 10
 210 GO SUB 8000
 215 FOR h=1 TO 5
 220 PRINT AT 4,a;" ● "
 230 LET a=a+(INKEY$="p" AND a<2
9)-(INKEY$="o" AND a>0)
 240 IF INKEY$="m" THEN GO SUB 1
000
 490 NEXT h
 500 NEXT f
 550 GO TO 200
1000 PRINT AT 4,a+1;" "
1005 LET bh=0
1010 FOR n=5 TO 20
1020 IF ATTR (n,a+1)=112 THEN LE
T bh=1
1025 PRINT PAPER 6;AT n,a+1;"●"
1026 BEEP .025,0-n
1027 PRINT PAPER 6; INK 2;AT n,a
+1;"╤"
1030 IF bh=1 THEN GO TO 1500
1040 NEXT n
1050 NEXT n
1070 RETURN
1500 LET sc=sc+10
1560 FOR j=1 TO 10
1570 IF b(j,2)=a+1 THEN GO TO 15
90
1580 NEXT j
1590 LET b(j,1)=18
1595 LET b(j,2)=INT (RND*29)+1
1600 LET b(j,3)=b(j,3)+1
1610 PRINT AT 2,20;"PUNTOS=";sc
1620 PRINT PAPER 6; INK 3;AT b(j
,1),b(j,2);CHR$ b(j,4)
1630 RETURN
7000 FOR f=1 TO 10
7010 LET b(f,2)=INT (RND*29)+1
7020 LET b(f,1)=18: LET b(f,3)=1
7021 LET b(f,4)=147+INT (RND*2)
7025 PRINT PAPER 6: INK 0;AT b(f
,1),b(f,2);CHR$ b(f,4)
7030 NEXT f
7040 RETURN
8000 PRINT PAPER 6; INK 2;AT b(f
,1),b(f,2);"╤"
8005 BEEP .025,-30
8010 LET b(f,1)=b(f,1)-b(f,3)
8015 IF b(f,1)<=4 THEN GO TO 900
0
8016 IF b(f,4)=147 THEN LET b(f,
4)=148: GO TO 8020
8017 IF b(f,4)=148 THEN LET b(f,
4)=147
8020 PRINT PAPER 6; INK 0;AT b(f
,1),b(f,2);CHR$ b(f,4)
8040 RETURN
9000 PRINT FLASH 1;AT 4,b(f,2);"
"
9010 PRINT FLASH 1;AT 15,10;"MAL
A SUERTE"
9015 PRINT AT 21,0;"OTRO ASALTO?
 PULSE UNA TECLA..."
9020 BEEP .05,INT (RND*50): BORD
ER INT (RND*8): IF INKEY$="" THE
N GO TO 9020: PAUSE 0
9030 GO TO 60
9600 FOR f=0 TO 7: READ s: POKE
USR "b"+f,s: NEXT f
9610 DATA 255,16,16,16,255,1,1,1
9620 FOR f=0 TO 7: READ s: POKE
USR "d"+f,s: NEXT f
9630 DATA 0,88,88,60,26,26,36,36
9640 FOR f=0 TO 7: READ s: POKE
USR "e"+f,s: NEXT f
9650 DATA 0,26,26,60,88,88,36,36
9660 FOR f=0 TO 7: READ s: POKE
USR "f"+f,s: NEXT f
9670 DATA 60,126,255,255,255,255
,126,60
9680 RETURN
```

LA SERPIENTE GLOTONA

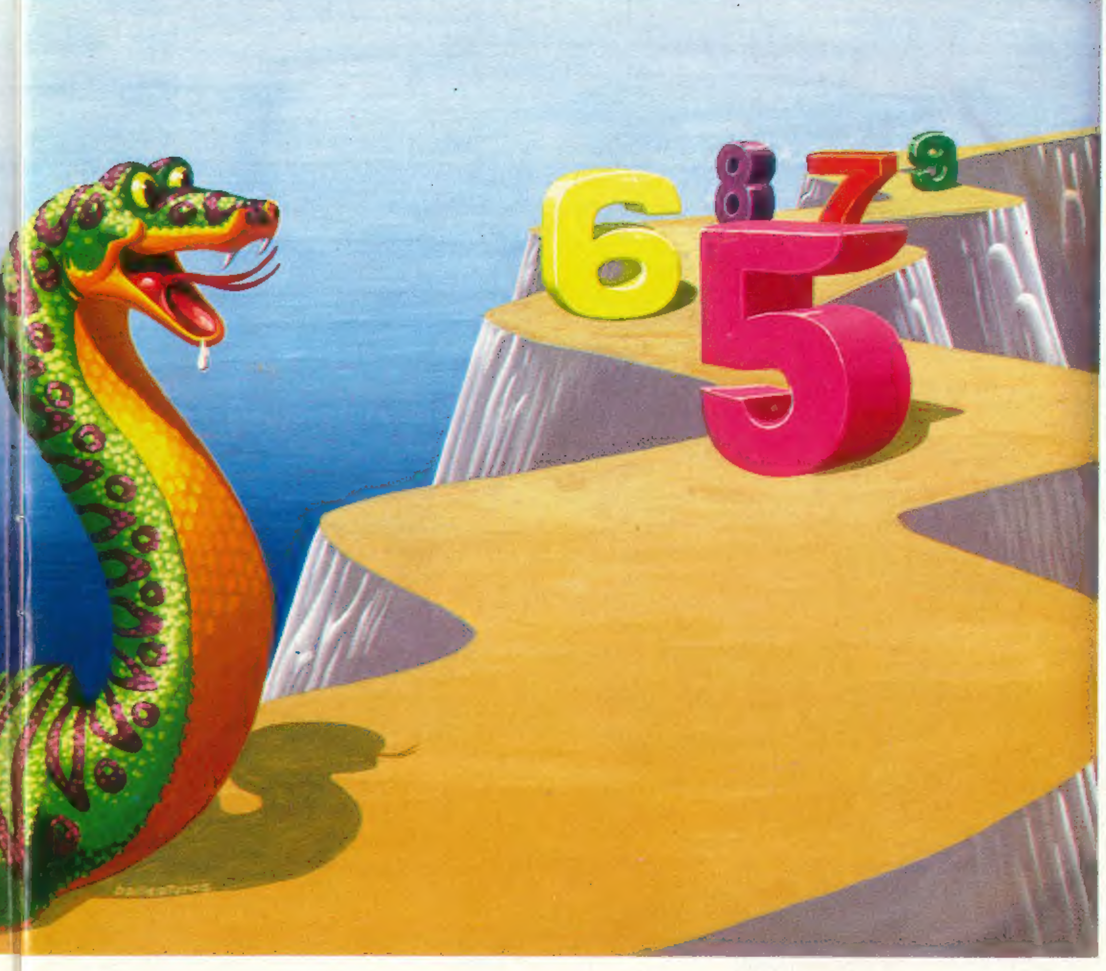
Spectrum 16 K

De todos es conocida la pesada digestión que padecen las serpientes tras sus almuerzos, sobre todo si éstos son copiosos. Pues bien, en esta ocasión podremos comprobar en la pantalla de nuestro ordenador todo este largo proceso, que va aumentando considerablemente a medida que la glotona serpiente va enguyendo más y más números, que en esta ocasión representan su alimento, hasta quedar paralizada.

Este programa, sin lugar a duda, es un juego y de los clásicos: se trata de comer la mayor cantidad de cosas en el mínimo tiempo posible.

Las «víctimas» son cuadraditos de colores con un número en su interior que aparecen aleatoriamente en la pantalla, y el actor principal, una serpiente hambrienta dirigida por usted.

Dispone de tres minutos de tiempo real (según muestra el reloj incorporado) para batir el récord de puntos que aparece en pantalla; según va devorando cuadraditos y transcurre el tiempo, la serpiente crece, dificultando considerablemente su libertad. de movimientos. Cuando la serpiente glotona consigue devorar todos los números, habrá superado la prueba.

La futura digestión no parece preocupar a la serpiente. Lo importante para ella es comer.

J.R. BALLESTEROS

Las teclas que le permiten moverse en las cuatro direcciones son:

—«Q» hacia ARRIBA.
—«A» hacia ABAJO.
—«P» hacia DERECHA.
—«O» hacia IZQUIERDA.

Eso sí, cuando bata el récord, el programa le preguntará su nombre (!).

```
NOTAS GRAFICAS
-----------------
A B C D E F
✤ ▷ △ ◁ ▽ ◆
```

```
   5 REM [illegible]
  10 GO SUB 9000: GO TO 1000
  20 DEF FN g()=INT ((PEEK 23672
+256*PEEK 23673+65536*PEEK 23674
)/50)
  30 LET ss=0: LET c$="XXXXXX"
  40 PAPER 6: INK 7: BORDER 1: C
LS
  50 LET t=180: LET s=0: LET a=1
0: LET b=10: LET f$="": LET z=1:
 LET m=5: LET a$="p": LET b$=a$:
 DIM x(35): DIM y(35): GO TO 170
  60 LET f=INT (RND*9)+1: LET d=
INT (RND*16)+3: LET e=INT (RND*2
7)+2
  70 IF SCREEN$ (d,e)<>" " THEN
GO TO 60
  80 IF SCREEN$ (d,e+2)<>" " THE
N GO TO 60
  90 IF SCREEN$ (d+1,e+1)<>" " T
HEN GO TO 60
 100 IF SCREEN$ (d+2,e)<>" " THE
N GO TO 60
 110 IF SCREEN$ (d+2,e+2)<>" " T
HEN GO TO 60
 120 PRINT AT d,e; INK RND*4;"##
#";AT d+1,e; INK RND*4;"#";f;"#"
;AT d+2,e; INK RND*4;"###": RETU
RN
 160 PRINT AT d,e;"   ";AT d+1,e
;"   ";AT d+2,e;"   ": RETURN
 170 PRINT INK 2;AT 0,1; FLASH 1
;"SNAKE"; FLASH 0: PRINT INK 2;A
T 0,25;"RECORD"
 180 PRINT INK 2;AT 2,0;"✤✤✤✤✤✤✤
✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤"
 190 FOR n=3 TO 20: PRINT AT n,0
; INK 2;"✤";AT n,31;"✤": NEXT n
 200 PRINT INK 2;AT 21,0;"✤✤✤✤✤✤
✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤✤"
 210 PRINT AT 1,1; INK 0;"PUNTOS
:";AT 1,11; INK 1;"TIEMPO:";AT 1
,20; INK 2;ss;AT 1,25 ;c$
 220 POKE 23674,255: POKE 23673,
255: POKE 23672,255: GO SUB 60:
GO TO 280
 240 LET z=z+1: IF z=m THEN PRIN
T AT x(z),y(z);" ": LET z=1: LET
 m=m+(1 AND m<35)
 250 LET t=FN g(): PRINT INK 2;A
T 1,20;t: IF t>=180 THEN GO TO 6
00
 260 BEEP .001,40
 270 PRINT AT a,b; INK RND*4;"◆"
: LET b$=a$: PRINT AT x(z),y(z);
" "
 280 IF INKEY$="" THEN LET b$=a$
 290 LET a$=INKEY$: IF a$<>"p" A
ND a$<>"o" AND a$<>"a" AND a$<>"
q" THEN LET a$=b$
 300 IF a$="o" THEN LET b=b-1: L
ET c=2
 310 IF a$="a" THEN LET a=a+1: L
ET c=3
 320 IF a$="q" THEN LET a=a-1: L
ET c=1
 330 IF a$="p" THEN LET b=b+1: L
ET c=0
 340 LET x(z)=a: LET y(z)=b
 350 IF SCREEN$ (a,b)="#" THEN G
O TO 400
 360 IF SCREEN$ (a,b)="" THEN GO
 TO 700
 370 IF RND>.98 THEN GO SUB 160:
 GO SUB 60
 380 PRINT AT a,b;CHR$ (145+c)
 390 GO TO 240
 410 PRINT AT a,b;CHR$ (145+c)
 420 LET s=s-1: INK 1: FOR n=f T
O 0 STEP -1: LET s=s+1: BEEP .00
8,n+30: PRINT AT 1,8;s: PRINT AT
 d+1,e+1;n: NEXT n
 448 GO SUB 160: PRINT AT a,b;CH
R$ (145+c): FOR n=0 TO 20: NEXT
n: GO SUB 60: GO TO 240
 600 FOR n=60 TO 30 STEP -1: BEE
P .1,n: NEXT n
 605 PRINT AT 2,0; FLASH 1; INK
2;"GAME OVER"
 610 IF s>ss THEN PRINT AT 3,0;
FLASH 1;"NUEVO RECORD": LET ss=s
: GO TO 630
 620 GO TO 650
 630 INPUT "Ponga su nombre";c$:
 IF LEN c$>7 THEN GO TO 630
 650 INPUT "Otra vez?";d$: IF d$
="s" OR d$="S" THEN GO TO 40
 660 GO TO 10
 700 PRINT AT a,b;CHR$ (145+c)
 710 FOR n=30 TO 1 STEP -1: BEEP
 .1,n: NEXT n: GO TO 650
1000 BRIGHT 1: PAPER 6: INK 2: B
ORDER 6: CLS
1020 PRINT AT 2,12; BRIGHT 1; FL
ASH 1;"SNAKE";AT 3,12;"[illegible]"
1040 PRINT AT 5,8;"INSTRUCCIONES
";AT 5,8; OVER 1;"_____________"
1060 PRINT AT 11,8;"“SNAKE” debe
 morder las  casillas numeradas,
 obteniendo  la puntuacion indic
ada."
1070 PRINT AT 15,8;"CONTROL"'''"
(q)= arriba","  (o)= izquierda"'"
 (a)= abajo","  (p)= derecha"
1090 PRINT INK 0;AT 21,0;" PULSE
 UNA TECLA PARA CONTINUAR": PAUS
E 0
1100 LET e=26: LET d=7: LET f=9:
 GO SUB 70
1110 LET x$="                   ◆
◆◆◆◆◆◆◆▷": FOR n=24 TO 0 STEP -1
: BEEP .001,40: PRINT AT 8,0;x$(
n+1 TO 27): PAUSE 4: NEXT n
1130 FOR n=f TO 0 STEP -1: BEEP
.008,n+30: PRINT AT d+1,e+1;n: N
EXT n
1140 PAUSE 100
1150 FOR n=7 TO 21: PRINT ;AT n,
0;"
   ": NEXT n
1160 PRINT INK 2;AT 7,5;"El tiem
po limite del juego  es de “3” m
inutos,indicados en  segundos."
1170 PRINT INK 2;AT 14,5;"SNAKE
crece continuamente,  si se enci
erra, se muerde o se  da con la
pared muere."
1180 PRINT INK 0;AT 21,0;" PULSE
 UNA TECLA PARA CONTINUAR"
1230 PAUSE 0: GO TO 30
9000 RESTORE : FOR n=144 TO 149:
 FOR f=0 TO 7: READ a: POKE USR
CHR$ (n)+f,a: NEXT f: NEXT n: RE
TURN
9150 DATA 8,16,8,186,93,16,8,16,
192,156,130,129,129,130,156,192,
24,36,66,66,66,0,129,255,3,57,65
,129,129,65,57,3,255,129,0,66,66
,66,36,24,24,36,90,189,189,90,36
,24
9999 BORDER 1: PAPER 4: INK 0: B
RIGHT 1
```

LOS MAS VENDIDOS

1. Sabre Wulf	Ultimate
2. Fighter Pilot	Digital
3. Decathlon	Ocean
4. Full Throttle	Micromega
5. TLL	Vortex
6. Match Point	Psion
7. Olimpicom	Mitec
8. Beach Head	U.S. Gold
9. Atic Atac	Ultimate
10. Psitrom	Beyond

LISTA ELABORADA CON LA COLABORACION DE:

Chips and Tips
Micro World
Gesco Informática
World Micro
Epsilom
Tronik

ROBOT FACTORY

Ventamatic
48 K
Tipo de juego: Arcade
P.V.P.: 1.700

Se trata de un juego dentro de la más pura línea de los programas de acción, entendiendo por éstos, aquellos que hay que controlar, usando un determinado número de teclas, y en los que no se permite el mínimo fallo.

El objetivo principal del juego es conseguir fabricar diez piezas sin defectos, en cada una de las habitaciones por las que hay que pasar para cumplir el ciclo completo del juego.

Estas son diferentes cada vez y presentan unas dificultades que son también distintas según la pantalla en la que nos encontremos.

Cada una tiene un determinado número de pisos por los que hay que hacer caer las piezas que aparecen aleatoriamente en la parte superior de la imagen, pero al hacerlo, es necesario conseguir que éstas pasen a través de las máquinas de producción.

Como en todos los juegos, existe un enemigo que intentará hacer fracasar nuestra misión, en esta ocasión se trata de piezas defectuosas, de las cuales tendremos que defendernos y además evitar que destruyan a las que no lo son.

La presentación del juego, con una melodía agradable, resulta muy pesada después de llevar un rato con ella y las instrucciones presentadas en forma de scroll horizontal, son demasiado lentas.

Las diferentes pantallas que componen el juego son originales y algunas están llenas de imaginación. Los gráficos están bien, sin embargo, el movimiento resulta un tanto lento para este tipo de juegos. Otro inconveniente lo supone el hecho de que una vez que empezamos a jugar, el personaje que dirigimos no cesa de moverse en ningún momento, salvo si pulsamos una tecla constantemente, lo que resulta bastante incómodo.

El movimiento se rige por los cursores, motivo éste por el que no resulta complicado su control.

Un juego que quiere hacernos recordar a Manic Miner, salvando las distancias, que en este caso son muchas.

Entretenido, difícil y con rasgos de originalidad, resulta apto para pasar un rato agradable tratando de salvar los obstáculos a los que continuamente habrá que enfrentarse.

AVALON

Hewson Consutants/ERBE
48 K
Tipo de juego: Aventura Gráfica
P.V.P.: 2.100



Avalon es un juego de reciente aparición en Inglaterra, que viene a poner una de las primeras piedras en lo que ya empieza a ser una nueva forma de programación. Este tipo de aventuras viene a sustituir de algún modo a aquellas otras gráfico-conversacionales, ya que a pesar de estar basados en un tipo de desarrollo parecido al de éstos, incorporan además el factor de movimiento, lo que sin lugar a dudas da una mayor vistosidad al juego y lo hace.

Nuestra misión consiste en guiar a Maroc (el mago) a través de más de 200 habitaciones, túneles y cuevas, para lo cual hay que pasar por ocho niveles distintos. El objetivo prioritario es conseguir destruir al Señor del Caos.

El mago dispone de una serie de hechizos que puede utilizar según le convenga, unas veces para defenderse, y otras para descrifrar enigmas que le abran puertas imposibles de atravesar normalmente.

También se pueden recoger algunos objetos, simplemente pasando por encima de ellos. Los habrá que nos sirvan como ayuda para completar la aventura, y otros, que nos sean totalmente inútiles.

En nuestro recorrido, iremos pasando, según avancemos, por barracones y sala de guardias, cuevas naturales en forma de habitaciones, cuevas recortadas que alimentaban las forjas del Señor del Caos, grandes salones, laberintos y catacumbas.

Los hechizos de que disponemos son muchos y con diferentes posibilidades de utilización, según las fases del juego en las que nos encontremos.

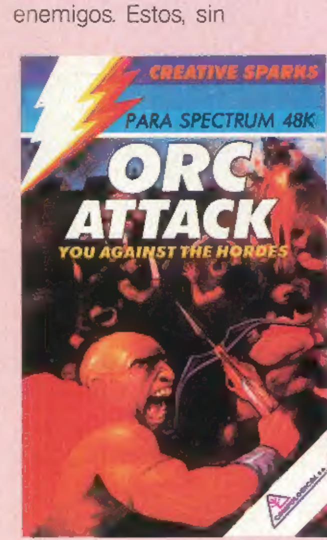
Todo el juego está realizado de forma muy original, tanto los gráficos como la ambientación son muy buenos. El movimiento quizás pueda resultar un tanto complicado al principio sobre todo a la hora de abrir las puertas de las cuevas, pero en seguida se le acaba cogiendo el tranquillo al asunto, y es entonces cuando el juego puede resultar enormemente atractivo.

El diseño de las distintas pantallas, los gráficos tridimensionales y la originalidad del juego, hacen de éste un buen motivo para pasar un rato entretenido, tratando de conseguir llevar a cabo la nada fácil misión encomendada en el juego.

ORC ATTACK

Creative Sparks/Compulogical
48 K
Tipo de juego: Arcade
P.V.P.: 1.500

El juego consiste en defender un castillo del ataque de nuestros enemigos, para lo cual contamos con piedras, espadas o aceite hirviendo, según las diferentes fases del juego. Al principio, por ejemplo, utilizamos piedras, y cuando los atacantes logran subir hasta arriba, tenemos que defendernos con la espada. Disponemos de nueve vidas, las cuales vamos perdiendo a medida que nos alcanza alguna flecha o que somos derrotados por algún asaltante. Cuando el aceite hirviendo esté preparado, suponiendo que logremos llegar hasta entonces, podremos destruir a los enemigos. Estos, sin



embargo, pueden resucitar con el hechizo de un brujo, que es su aliado, y además intentarán escalar nuestra muralla.

El programa es bastante simple; los gráficos aunque sencillos en su realización, cumplen perfectamente con el cometido del juego. El movimiento no resulta demasiado eficaz, sobre todo a la hora de intentar pararnos para detener el ataque enemigo desde una posición determinada.

A pesar de su sencillez, se trata de un juego bastante complicado y lograr conseguir el control de la situación nos va a suponer un notable esfuerzo.

Es un juego sin complicaciones, en el que no es necesario romperse la cabeza. Recomendado especialmente para los amantes de los juegos del tipo arcade.

1 X 2

Microgesa
48 K
Utilidades
P.V.P.: 3.900

Este programa nos permite realizar el desarrollo completo de un boleto múltiple con todos los dobles y triples que deseemos. El boleto debe, sin embargo, reducirse, ya que en todas las apuestas siempre existe un buen porcentaje de combinaciones que lo componen y que tienen escasa probabilidad de salir, de ahí que se utilicen unos parámetros estadísticos que nos ayudarán a confeccionar una lista de probabilidades que limiten el tipo de desarrollo.

Las combinaciones pueden imprimirse eligiéndose con anterioridad en grupos de 8 apuestas para cada boleto. Si se utiliza una impresora ADMATE, éstos se rellenarán de acuerdo a las normas del patronato; para ello, basta con cuadrar adecuadamente el impreso correspondiente.

El programa ofrece además una serie de opciones como son rellenar un impreso múltiple o hacer el escrutinio. Esta última opción comprueba rápidamente sus aciertos.

Para ello es necesario tener en memoria las apuestas que vamos a comprobar y llenar seguidamente el boleto que contiene el pleno de 14 aciertos. También es posibe grabar en cinta todas las apuestas que se encuentren en memoria.

Hay una posibilidad, incorporada en el programa, que permite eliminar el número de apuestas que se desee de forma aleatoria.

Se trata, en definitiva, de un programa que puede resultar muy interesante para todos los aficionados a las quinielas y aquellos que pretenden ahorrarse algún dinero a la hora de hacer sus combinaciones.

FULL THROTTLE

Micromega/ABC
48 K
Tipo de juego: Arcade
P.V.P.: 1.695



En esta misma sección hemos hablado con anterioridad de un programa que simulaba una carrera de coches por diferentes circuitos, en esta ocasión la competición la llevamos a cabo utilizando motos de carreras.

El programa, que ha sido confeccionado en tres dimensiones, permite escoger entre diez circuitos diferentes, que reproducen fielmente los diez mejores del mundo (incluido el de el Jarama). Una vez que tenemos el juego en pantalla, se nos ofrecen cinco opciones: elegir uno de los diez circuitos, elegir el número de vueltas, practicar, elegir el teclado o joystick y comenzar a jugar.

Al principio, es conveniente practicar durante algún tiempo para lograr la destreza suficiente que nos permita afrontar con unas mínimas garantías la prueba.

Cuando comienza la competición corremos junto a cuarenta pilotos más. Nosotros empezamos siempre en la última posición, por lo que tenemos que intentar escalar todos los puestos posibles para lograr acabar los primeros. Esto, sin embargo, nos va a resultar muy difícil, ya que el resto de los corredores se encuentran apelotonados y chocar con ellos suele ser bastante fácil si no tenemos la suficiente destreza, lo que va a suponer, además, que nuestra velocidad vuelva a cero.

El programa, desde el punto de vista de la perspectiva y la capacidad gráfica, resulta correcto, la inclinación del motorista cada vez que vamos a un lado o a otro es muy vistosa, y los efectos sonoros a nivel de motor y derrapaje resultan espectaculares. La presentación es buena, pero no podemos decir lo mismo del colorido, que desde el punto de vista estético no parece el más apropiado.

El nivel de dificultad es muy alto y cualquier fallo puede alejarnos del resto de los corredores. Por lo demás, se trata de un buen juego, en

el que la velocidad y el riesgo van a ser los denominadores comunes.

BC BILL

Imagine/ABC
48 K
Tipo de juego: Arcade
P.V.P.: 1.495

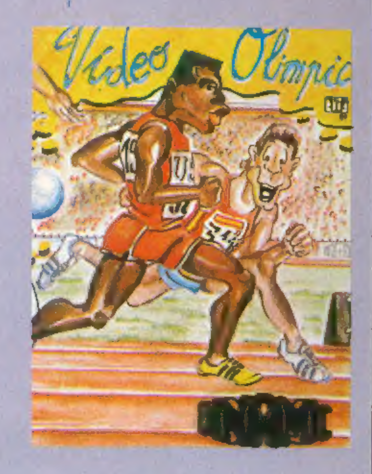


Las notas de humor en los programas comerciales, además de ser una sana costumbre, dota a éstos de un atractivo especial. BC Bill es un programa lleno de originalidad en todos sus detalles que nos hará sonreír cuando juguemos con él.
Bill, es un curioso troglodita que tiene que mantener viva a su especie. Para ello ha de conseguir un determinado número de mujeres que le permitan formar una gran familia; pero cuidado, el tiempo es limitado y es necesario tener, al menos, una mujer que nos proporcione descendencia. Otro dato a tener en cuenta es que nuestra familia necesita estar alimentada para sobrevivir, por lo que es necesario practicar el noble oficio de la caza. Como en todos los juegos de este tipo, tenemos unos enemigos, que en este caso se trata de dinosaurios y una especie de pajarracos de color negro que hay que tratar de evitar y a la vez impedir que acaben con nuestra especie. Una forma de evitar el ataque de los dinosaurios es ofrecerle tres animales; de esta forma nos dejará tranquilo.
Si no conseguimos alimentar a nuestras mujeres, éstas desaparecerán y nuestros hijos abandonarán la cueva, pero si, por el contrario, todo sale bien, nuestros hijos alcanzarán la mayoría de edad y podrán formar su propia familia.
El juego en líneas generales está bien conseguido, con una presentación elaborada a la que acompaña una serie de melodías bastante buenas, que según nos ponen en la carátula han sido creadas por el famoso compositor de Jazz, Abdul Ibrahim.
Los gráficos están dentro de la misma línea humorística del resto del programa, por lo que se acoplan perfectamente al sentido del juego. Por lo que se refiere a la composición del teclado en relación al movimiento, es necesario hacer hincapié en la dificultad que entraña su manejo, ya que a pesar de realizar sólo cinco movimientos, el acceso a éstos es complicado.
A pesar de su aparente sencillez es un juego que resulta muy completo y divertido. Recomendado para aquellos que no quieren complicarse la vida y desean pasar un rato entretenido.

VIDEOLIMPIC

Dinamic/Micro World
48 K
Tipo de juego: Simulador deportivo
P.V.P.: 1.800

Continuando con la saga de juegos deportivos que han venido apareciendo en los últimos días en nuestro país, nos llega en esta ocasión una producción española de reciente aparición en el mercado, Videolimpic.
El programa reproduce con bastante fidelidad el desarrollo de una serie de competiciones deportivas, como son: los 100 metros lisos, el salto de longitud, lanzamiento de jabalina, lanzamiento de martillo, 100 metros valla y natación.
La presentación del juego es muy buena, la ambientación está bien conseguida y los gráficos cumplen perfectamente con las necesidades del programa.
A diferencia de otros juegos de este tipo, utiliza un sistema de movimiento de gráficos que ha sido realizado carácter a carácter, en vez de pixel a pixel, consiguiendo de esta forma que el corredor

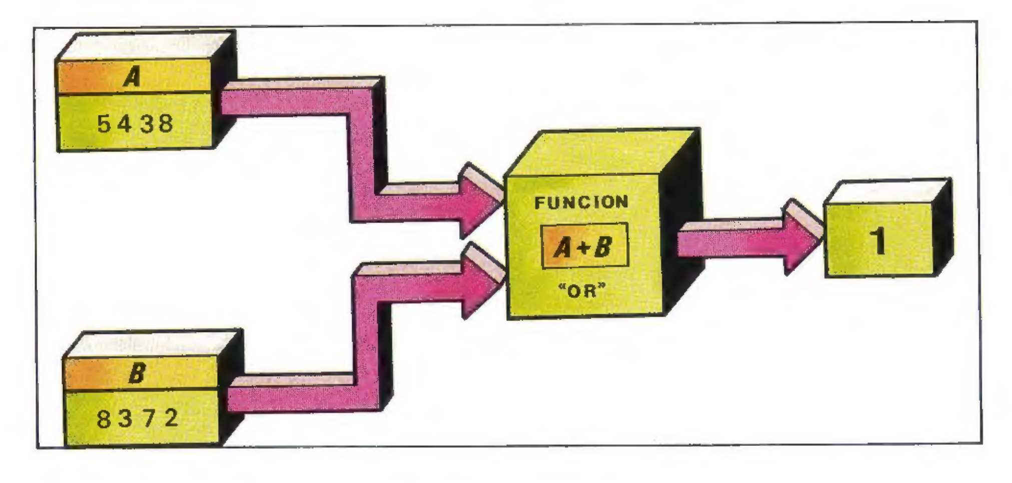


avance rápidamente, dándonos una sensación mayor de velocidad.
El nivel de dificultad es alto, por lo que se requiere una destreza suficiente para conseguir ir superando cada una de las pruebas, y hay que tener cuidado, porque si no superamos alguna, tendremos que comenzar otra vez desde el principio.
La marca a superar en cada competición nos aparece en el marcador situado en la parte superior de la imagen, donde, además, nos ofrecen toda la información referente a las marcas conseguidas.
Cada vez que se completa un ciclo del juego (superamos con éxito todas las pruebas), pasamos al pódium, donde se nos otorga la medalla correspondiente y escuchamos el himno nacional. Después de esto volveremos a competir, pero en esta ocasión con unas marcas más difíciles. Si conseguimos superar varios ciclos, nos llevaremos una sorpresa al llegar al pódium.
El juego, que es muy entretenido y está bien hecho, es especialmente recomendado para los que no tengan tiempo de hacer deporte y quieran batir récords desde la butaca de su salón, luchando con el teclado y emulando a los grandes campeones olímpicos.

■■■ *Ya está disponible en nuestro país* ***Wafadrive****, una unidad de microdrive doble, con mayor capacidad de almacenamiento que los habituales, cuyo precio oscilará en torno a las 40.000 ptas.*

■■■ ***Microgesa*** *tendrá en breve disponible en el mercado unos programas técnicos para microdrive:* ***Agente de bolsa, Mediciones y presupuestos*** *y* ***Administración de fincas.***

■■■ *Ya está disponible en nuestro país el* ***Match Point****, la última creación de la casa* ***Psion****, que ha tenido un gran éxito en Inglaterra. El programa, a juicio de los expertos, es la mejor creación de esta compañía.*



teres que ocupan dos posiciones cada uno, estos son los correspondientes a los símbolos " < =", " > =", y " <> ".

Transmisión del ASCII

En el lenguaje BASIC del Spectrum, la transmisión de caracteres ASCII puede efectuarse o directamente en este código o anteponiendo la sentencia "CHR $" al correspondiente código decimal.

Ejemplos:

ASCII	DECIMAL
PRINT «A»	PRINT CHR$ 65
PRINT «a»	PRINT CHR$ 97

El primer método ofrece ciertas ventajas con respecto al segundo.

- Es legible directamente.
- Ocupa menos sentencias.
- Más rápido de ejecución.

si por el contrario todo lo que desea es semi-camuflar el mensaje, será conveniente utilizar el segundo.

Ejecute la siguiente instrucción directa y compare el resultado con el proporcionado por el programa número «3».

PRINT «Maese Pérez el organista»

PROGRAMA 3

```
10 REM ******************
       *                *
       *     CURSO      *
       *                *
       * BASIC/SINCLAIR *
       *                *
       *     "CHR$"     *
       *                *
       ******************

20 BORDER 7: PAPER 7: INK 0: C
LS

30 REM ****************
       *              *
       * IMPRESION DE *
       *              *
       *  CARACTERES  *
       *              *
       ****************

35 RESTORE
40 FOR X=1 TO 24
50 READ codigo decimal
60 PRINT CHR$ codigo decimal;
70 NEXT x
75 PRINT
80 REM
       **************
       *            *
       *  TABLA DE  *
       *            *
       * CARACTERES *
       *            *
       **************

90 DATA 77,97,101,115,101,32,8
0,101,114,101,122,32,101,108,32,
111,114,103,97,110,105,115,116,9
7
```


OPERADORES

Los operadores son símbolos que expresan el tipo de operación que ha de realizarse, bien entre dos constantes, bien entre una variable y una constante, etc. Veamos unos ejemplos.

OPERANDO	OPERADOR	OPERANDO
3	+	12ØØ
valor	=	1ØØ
A$	<>	«Antonio»
527	AND	32

Existen tres tipos de operadores:

- ARITMETICOS.
- DE RELACION.
- LOGICOS.

Operadores aritméticos

Son aquellos que permiten ejecutar las operaciones aritméticas básicas: suma, resta, etc. Los símbolos utilizados son los indicados en la tabla.

OPERACION	SIMBOLO
Suma	+
Resta	—
Multiplicación	*
División	/
Potenciación	↑

Ejemplos:
1ØØ2 * 7 (1ØØ2 por 7)
3 ↑ 2 (3 al cuadrado)
456 / 2 (456 entre 2)

Expresiones aritméticas

Son conjuntos de constantes y variables unidas entre sí por operadores aritméticos.

Ejemplos:

(3 + 5) / 7
8 ↑ 2 / 1Ø * 5
gr * X ↑ 3 /tz
voltaje ↑ 2 / resistencia * tiempo

Las variables utilizadas tienen que estar definidas previamente, ya que de lo contrario, cuando el ordenador realice un cálculo y alguna no lo esté, enviará el siguiente mensaje de error:

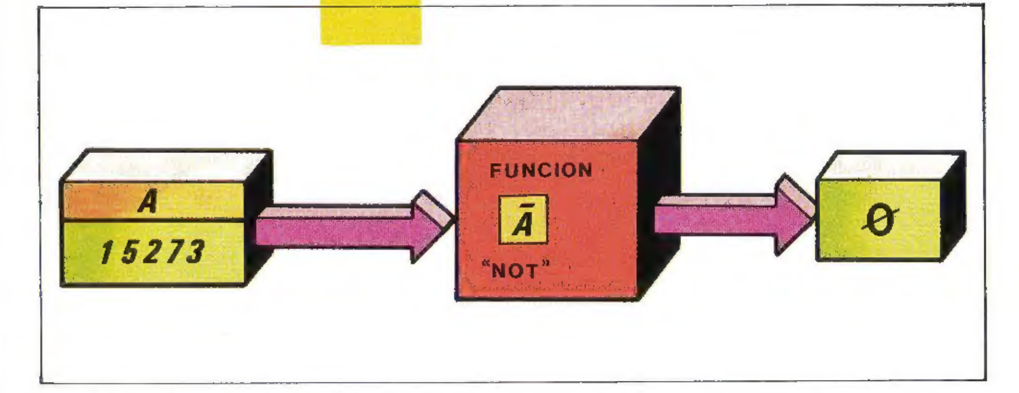
2 Varible not found

Al igual que en álgebra, se pueden utilizar los paréntesis, pero no las multiplicaciones implícitas, es decir, la expresión (5 + X) (8 — Y), en BASIC, se escribe:

(5 + X) * (8 — Y)

Cálculo de expresiones

El ordenador cuando realiza un cálculo, lo hace siempre atendiendo al valor de *prioridad* que tenga cada operación. De las cinco operaciones aritméticas básicas, la potenciación tiene mayor prioridad que el resto, después la multiplicación y la división, ambas con el mismo valor, y por último, la suma y la resta, también con idéntica prioridad.



Función «AND». Estructura 1.

Cuando en una expresión hay dos operaciones de la misma prioridad, el ordenador efectúa los cálculos de izquierda a derecha.

Veamos por pasos cómo el ordenador calcula la siguiente expresión para x = 3 e y = 2Ø:

16 + x * 3Ø — y / 2
a) 16 + 3 * 3Ø — 2Ø / 2
b) 16 + 9Ø — 2Ø /2
c) 16 + 9Ø — 1Ø
d) 1Ø6 — 1Ø
Resultado: 96

Utilizando los paréntesis se puede alterar el orden de evaluación de las operaciones, ya que éstos se calculan primero. Aprovechando la expresión anterior, vamos a observar que el resultado varía colocando los paréntesis en las operaciones de menos prioridad.

(16 + x) * (3Ø — y) / 2
a) (16 + 3) * (3Ø — 2Ø) / 2
b) 18 * (3Ø — 2Ø) / 2
c) 18 * 1Ø / 2
d) 18Ø / 2
Resultado: 9Ø

Operadores de relación

Permiten realizar las comparaciones entre operandos (constantes o variables) tanto numéricos como de cadena.

OPERADOR	SIMBOLO
Igual	=
Distinto	<>
Mayor	>
Menor	<
Mayor o igual	>=
Menor o igual	<=

Las operaciones realizadas con estos operadores sólo tienen dos resultados o soluciones, es decir:

— Si la condición impuesta por el operador se cumple, es decir, que es *verdadera* (true), el valor del resultado es «1».

— Si por el contrario, la condición no se cumple, es decir, que es *falsa* (false), el valor se vuelve «Ø».

Ejecute los siguientes comandos directos y comprobará lo explicado anteriormente.

COMANDO	CONDICION	RESULTADO
PRINT 3 = 7	Falsa	Ø
PRINT 1ØØ>99	Verdadera	1
PRINT 8<> 8	Falsa	Ø
PRINT 1Ø>= 7	Verdadera	1

El símbolo «=» también sirve para asignar un valor a una variable.

Operadores lógicos

Se utilizan para realizar las operaciones lógicas a nivel de expresión.

OPERACION	FUNCION
Producto lógico	AND
Suma lógica	OR
Negación	NOT

En otras ocasiones, estos operadores se utilizan para relacionar dos expresiones mediante una condición, por ejemplo:

— Que el ordenador realice una determinada tarea si se cumplen varias relaciones.

RELACION	OPERADOR	RELACION
x >= Ø	AND	x <= 9

En este caso sólo se cumplirá la condición cuando la variable *x* sea mayor o igual a Ø y a la vez sea menor o igual a 9, es decir, que esté comprendida entre Ø y 9.

— Que se realice la tarea cuando simplemente alguna de las condiciones se cumpla.

RELACION	OPERADOR	RELACION
a= 1ØØ	OR	t > 1

La condición se cumple, bien cuando el valor de la variable *a* sea 1ØØ, bien cuando el valor de *t* sea mayor que 1 ó bien cuando se cumplan ambas condiciones.

Función «AND»

La estructura de esta función es la que se muestra a continuación, siendo a y b dos expresiones numéricas.

a AND b

Estas expresiones solamente pueden tomar los valores Ø o distinto de Ø, este último le vamos a representar como «<>Ø». Teniendo en cuenta todas las posibles combinaciones que pueden tomar a y b, vamos a mostrar su *tabla de verdad:*

a	b	RESULTADO
Ø	Ø	Ø
Ø	<>Ø	a (Ø)
<>Ø	Ø	Ø
<>Ø	Ø	a

de ésta se deduce que si el valor de *b* es igual a Ø, el resultado de la función es Ø, independientemente del valor de *a*. Si el valor de *b* es distinto de Ø, la función asume el valor de *a*.

Ejemplos:

OPERACION	RESULTADO
132 AND Ø	Ø
8Ø AND 1Ø	8Ø
Ø AND 25	Ø
1 AND 9Ø	1

La función AND, también puede tomar la estructura:

a$ AND b

en este caso el resultado de la función será una cadena vacía cuando *b* sea Ø y será la cadena *a$* cuando *b* sea distinto de Ø.

Ejemplos:

OPERACION	RESULTADO
«PEDRO» AND 3Ø	«PEDRO»
«JUAN» AND Ø	«»

Función «OR»

La estructura de la función *OR* es la siguiente:

a OR b

las expresiones *a* y *b* sólo pueden tomar valores numéricos. Con todas las combinaciones se obtiene la siguiente tabla de verdad:

a	b	Resultado
Ø	Ø	a (Ø)
Ø	<>Ø	1
<>Ø	Ø	a
<>Ø	<>Ø	1

Mirando detenidamente la tabla se observa que si *b* toma el valor Ø, la función resultante adquiere el de *a*, sin embargo, si el valor de *b* es diferente de Ø, el valor de la función es 1, independientemente del que tenga *a*.

Ejemplo:

OPERACION	RESULTADO
Ø OR 3Ø	1
2Ø OR Ø	2Ø
55 OR 7	1
Ø OR Ø	Ø

Función «AND». Estructura 2.

Función «NOT»

La función *NOT* afecta solamente a una expresión y su estructura es:

NOT a

su tabla de verdad es:

a	RESULTADO
Ø	1
<> Ø	Ø

Como se puede apreciar, el resultado de la función es la negación de la expresión *a*, es decir, vale 1 si *a* es igual a Ø y Ø si *a* es distinto de Ø. Ejemplo:

OPERACION	RESULTADO
NOT 7	Ø
NOT Ø	1

Anteponiendo el signo «—» a la función NOT, el resultado cambia de signo.

OPERACION	RESULTADO
— NOT 4	Ø
— NOT Ø	— 1

El programa «2» visualiza el código decimal correspondiente a la cadena ASCII introducida, de un máximo de 20 caracteres.

Organización del ASCII

Dentro del código ASCII, usado por el Spectrum, pueden encontrarse diversas zonas, teniendo cada una de ellas unas características distintas:

— CODIGO TRANSPARENTE.
— CODIGO ASCII CONVENCIONAL.
— CODIGO ASCII ESPECIFICO.
— GRAFICOS PREDEFINIDOS.
— GRAFICOS DE USUARIO.
— TOKENS.

El denominado *código transparente* está formado por una serie de comandos y funciones de control, como por ejemplo, el control de los cursores, el del color, la función EDIT, etc. Estos caracteres ASCII de control son específicos del Spectrum, otros ordenadores disponen de otro juego distinto.

OPERACION	SIMBOLO	PRIORIDAD	EJEMPLO
FRAGMENTACION	TO	12	"Juan" (1 TO 2)
POTENCIACION	↑	1Ø	1Ø ↑ 2
NEGACION	—	9	— 15
MULTIPLICACION	*	8	7 * 13
DIVISION	/		152 / 2
ADICION	+	6	1Ø + 4
SUBSTRACCION	—		8 — 5
OPERADORES RELACIONALES	= > < <> >= <=	5	1Ø 4
OPERADORES LOGICOS	NOT	4	NOT 5
	AND	3	1Ø AND 1
	OR	2	7 OR Ø

PRIORIDADES DE OPERACION

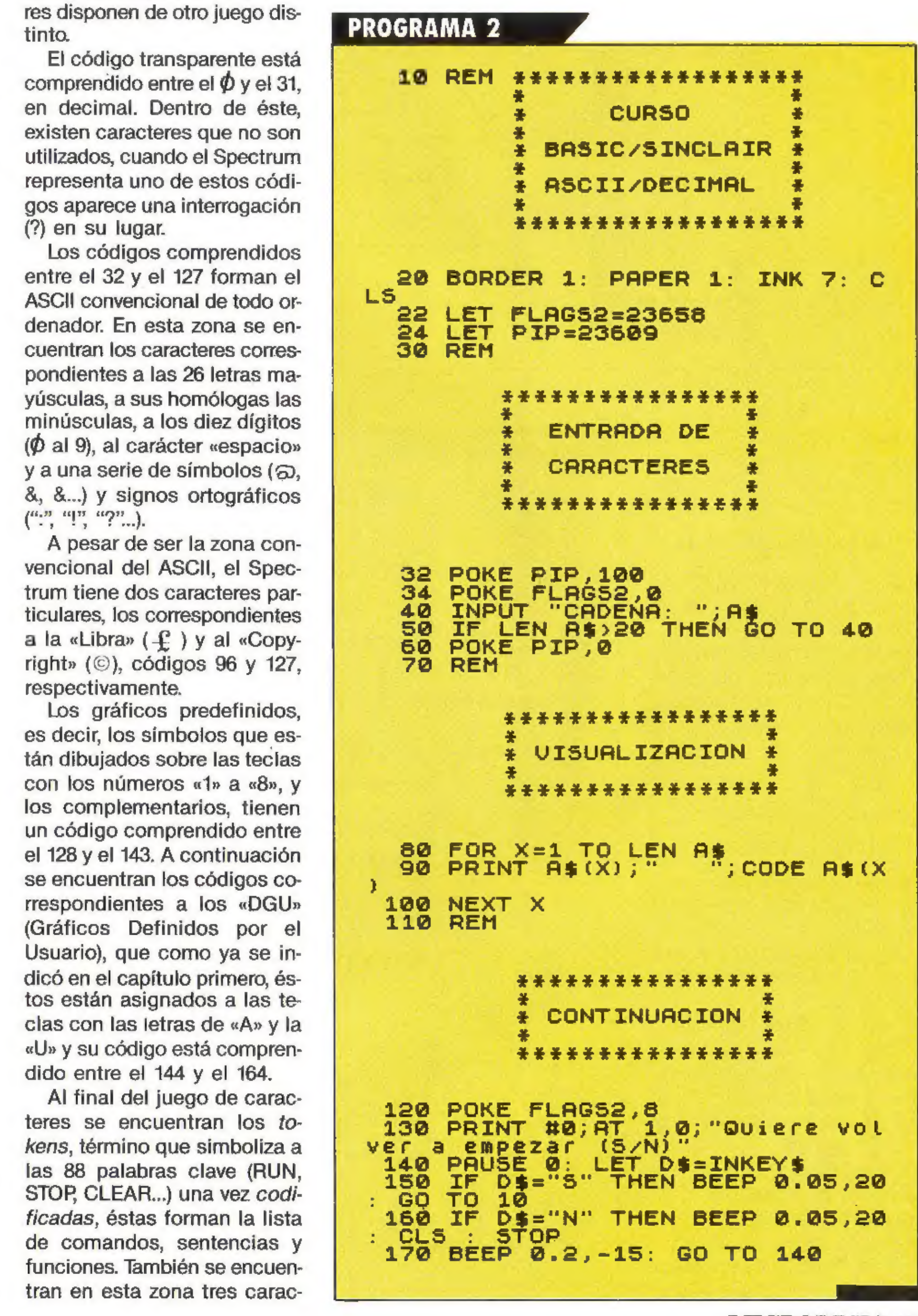
El código transparente está comprendido entre el Ø y el 31, en decimal. Dentro de éste, existen caracteres que no son utilizados, cuando el Spectrum representa uno de estos códigos aparece una interrogación (?) en su lugar.

Los códigos comprendidos entre el 32 y el 127 forman el ASCII convencional de todo ordenador. En esta zona se encuentran los caracteres correspondientes a las 26 letras mayúsculas, a sus homólogas las minúsculas, a los diez dígitos (Ø al 9), al carácter «espacio» y a una serie de símbolos (@, &, &...) y signos ortográficos (":", "!", "?"...).

A pesar de ser la zona convencional del ASCII, el Spectrum tiene dos caracteres particulares, los correspondientes a la «Libra» (£) y al «Copyright» (©), códigos 96 y 127, respectivamente.

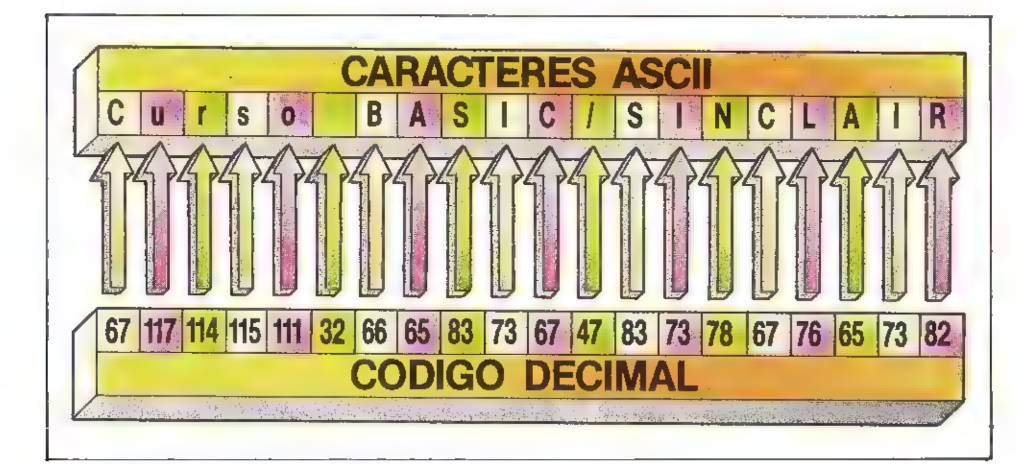
Los gráficos predefinidos, es decir, los símbolos que están dibujados sobre las teclas con los números «1» a «8», y los complementarios, tienen un código comprendido entre el 128 y el 143. A continuación se encuentran los códigos correspondientes a los «DGU» (Gráficos Definidos por el Usuario), que como ya se indicó en el capítulo primero, éstos están asignados a las teclas con las letras de «A» y la «U» y su código está comprendido entre el 144 y el 164.

Al final del juego de caracteres se encuentran los *tokens*, término que simboliza a las 88 palabras clave (RUN, STOP, CLEAR...) una vez *codificadas*, éstas forman la lista de comandos, sentencias y funciones. También se encuentran en esta zona tres carac-



PROGRAMA 2

```
  10 REM ******************
         *                *
         *      CURSO     *
         *                *
         * BASIC/SINCLAIR *
         *                *
         * ASCII/DECIMAL  *
         *                *
         ******************
  20 BORDER 1: PAPER 1: INK 7: C
LS
  22 LET FLAGS2=23658
  24 LET PIP=23609
  30 REM
        *****************
        *               *
        *   ENTRADA DE  *
        *               *
        *   CARACTERES  *
        *               *
        *****************
  32 POKE PIP,100
  34 POKE FLAGS2,0
  40 INPUT "CADENA: ";A$
  50 IF LEN A$>20 THEN GO TO 40
  60 POKE PIP,0
  70 REM
        *****************
        *               *
        * VISUALIZACION *
        *               *
        *****************
  80 FOR X=1 TO LEN A$
  90 PRINT A$(X);"   ";CODE A$(X
)
 100 NEXT X
 110 REM
        *****************
        *               *
        * CONTINUACION  *
        *               *
        *****************
 120 POKE FLAGS2,8
 130 PRINT #0;AT 1,0;"Quiere vol
ver a empezar (S/N)"
 140 PAUSE 0: LET D$=INKEY$
 150 IF D$="S" THEN BEEP 0.05,20
: GO TO 10
 160 IF D$="N" THEN BEEP 0.05,20
: CLS : STOP
 170 BEEP 0.2,-15: GO TO 140
```



Función «OR».

Ejercicio

Estas funciones pueden encadenarse para formar otra más complicada. Haciendo uso de los paréntesis «()» se consigue facilitar su interpretación; por ejemplo:

(X AND Y) OR (NOT X)

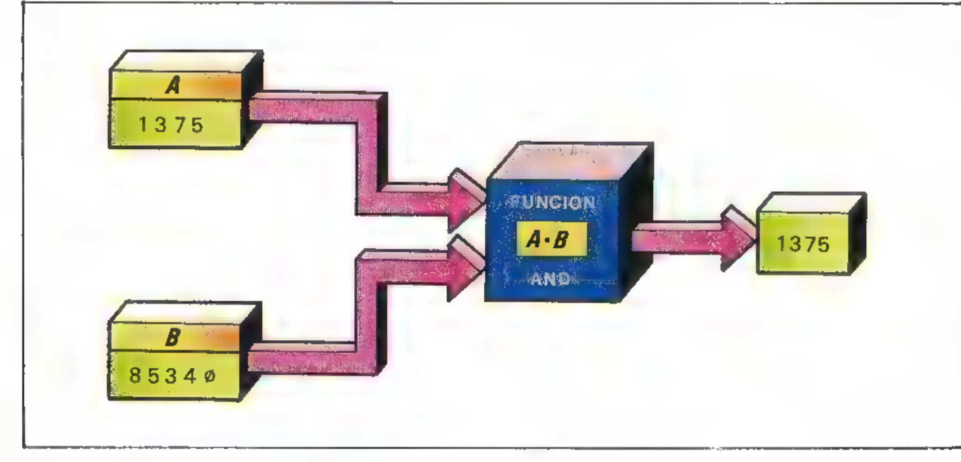
si asignamos a la variable X el valor 3 y a Y el valor 5, veamos cual es el resultado final resolviendo la función por pasos:

a) (3 AND 5) OR (NOT 3)
b) 3 OR (NOT 3)
c) 3 OR Ø
Resultado = 3

Como ejercicio, intente resolver la siguiente función:

— NOT ((X OR Y) AND (NOT Z))

Para los valores: X = 8, Y = 1Ø y Z = 83Ø. Si desea comprobar el resultado de su cálculo introduzca como comando directo la función anterior precedida de la sentencia PRINT.



Función «NOT».

CODIGO ASCII

Para representar un carácter en la pantalla del televisor o en la impresora, el ordenador utiliza el *código ASCII*; éste permite la transferencia de datos entre el ordenador y los dispositivos conectados a él (periféricos). Cada número, letra o símbolo tiene su representación en este código.

ASCII es la abreviatura, en inglés, de «American Standard Code for Information Interchange» que, traducido al idioma español, significa «Código normalizado Americano para intercambio de información».

El ASCII completo consta de 256 caracteres, cuyo código está comprendido entre Ø y 255.

Edite el programa número «1» que, una vez ejecutado, visualiza en pantalla los caracteres ASCII usados por el ZX Spectrum y comprendidos entre el código 32 y el 255.

Manejo de la tabla

Para conocer el *código decimal* correspondiente a un determinado caracter ASCII, basta con sumar los números de *fila* y *columna* indicados en la tabla.

código = fila + columna

Ejemplos:

CARACTER ASCII	NUMERO FILA	NUMERO COLUMNA	CODIGO
≠	3Ø	5	35
USR	19Ø	2	192
M	7Ø	7	77
8	5Ø	6	56

PROGRAMA 1

```
  10 REM ****************
         *              *
         *    CURSO     *
         *              *
         * BASIC/SINCLAIR *
         *              *
         *   "ASCII"    *
         *              *
         ****************

  20 BORDER 1: PAPER 1: INK 7: C
LS

  30 REM ***************
         *             *
         * IMPRESION DE *
         *             *
         *  CARACTERES  *
         *             *
         ***************

  40 FOR Y=32 TO 255 STEP 44
  50 FOR X=Y TO Y+43
  60 IF X<100 THEN PRINT " ";
  70 PRINT X;" ";CHR$ X,
  80 IF X=255 THEN GO TO 200
  90 NEXT X
 100 PRINT #0;AT 1,0;"Pulse una
tecla para continuar."
 110 PAUSE 0: BEEP 0.05,20
 115 CLS
 120 NEXT Y
 200 REM

         ****************
         *              *
         * CONTINUACION ? *
         *              *
         ****************

 210 POKE 23658,8
 220 PRINT #0;AT 0,0;"Desea obte
ner un nuevo listado  (S/N)"
 230 PAUSE 0: LET D$=INKEY$
 240 IF D$="S" THEN BEEP 0.05,20
: GO TO 10
 250 IF D$="N" THEN BEEP 0.05,20
: CLS : STOP
 260 BEEP 0.2,-15: GO TO 230
```



Periférico del año en UK.

¡Imagínate! Cinco utilísimos componentes integrados en una sola unidad compacta que se conecta a tu Spectrum en unos segundos. Sin cableados engorrosos. Potente y eficaz. Eso es **WAFADRIVE, elegido periférico del año en el Reino Unido y fabricado por Rotronics.** *

Interface con el Spectrum, interface serie RS/232 (con velocidad de transmisión seleccionable por software), interface paralelo Centronics y dos drives de **128 K cada uno** que utilizan cartuchos de 16, 64 o 128 K diseñados para proporcionar una **alta velocidad** de transferencia de datos (2 K por segundo) y la **máxima fiabilidad**.

Dale a tu Spectrum la potencia y la versatilidad de un sistema auténticamente profesional. Y aprovéchate de nuestra **Oferta Especial** de lanzamiento en la que, para que conozcas uno de los muchos programas ya disponibles para el **WAFADRIVE**, incluímos el **Spectral Writer** (un excelente Procesador de Textos). Y un cartucho virgen. Y manuales en castellano, claro.

Todo por 48.500 Ptas. en tu tienda de Informática o directamente en **MICROBYTE**.

* También fabricante de los moduladores ASTEC

Sí, quiero aprovecharme de su Oferta Especial de lanzamiento. Envíenme un WAFADRIVE, el Procesador de Textos Spectral Writer y un cartucho virgen, todo por 48.500.– Ptas.

Nombre

Dirección

Población Código Postal

Teléfono

Incluyo Talón Nominativo ☐
Contra Reembolso ☐

MICROBYTE San Gerardo, 59
28035 Madrid
Pedidos por teléfono:
91 - 656 50 02

M.H.

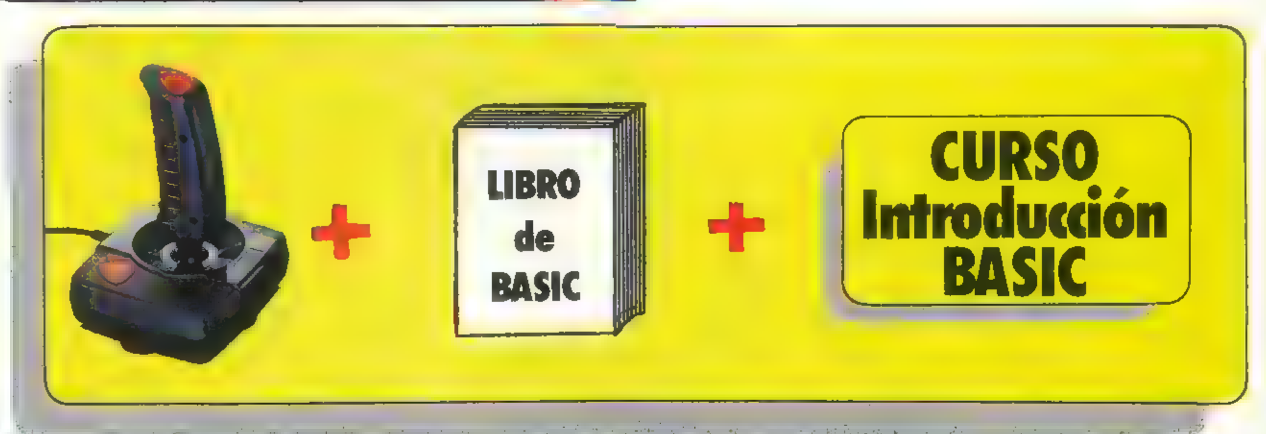
T&R

COMPUTIQUE

Te regala los 8 mejores programas



* Pssst
* Chess
* Chequered Flag
* Jet Pac
* Flight Simulation
* Reversi
* Cookie
* Backgammon



comprando un

ZX SPECTRUM 48 K



Por sólo 41.900 Ptas.

con la garantía Investrónica

Compras a plazo, hasta 24 meses

KEY INFORMATICA, S.A. Embajadores, 90 - 28012 MADRID - Teléfono: 227 09 80

Distribuidores oficiales de: sinclair commodore

DUELO A MUERTE

Manuel ORCERA VALERO, Madrid

Premiado con 15.000 Ptas.

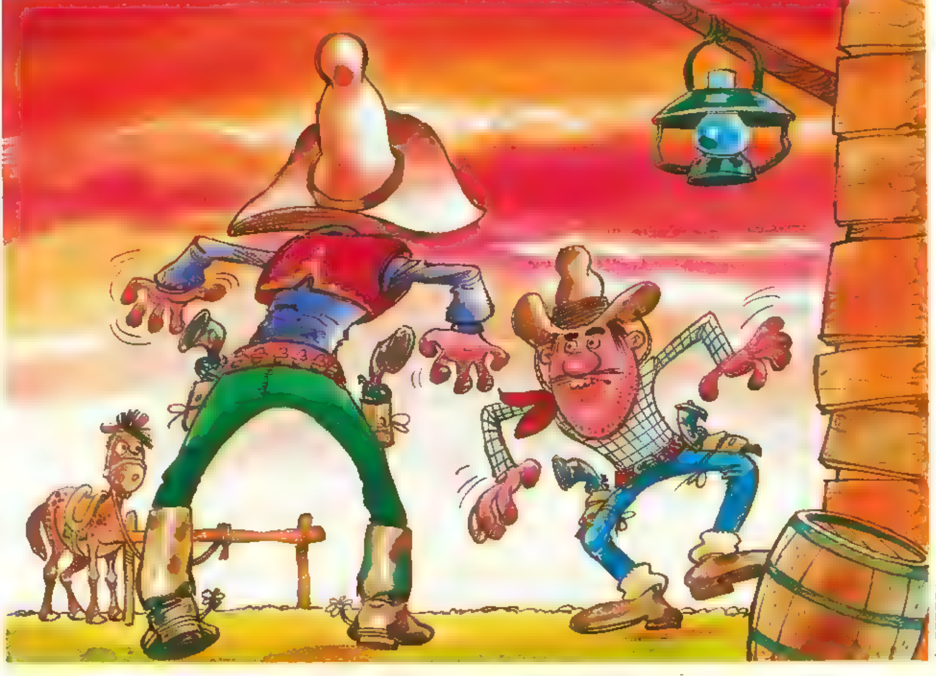
Spectrum 48 K

El Oeste polvoriento y rudo que todos conocemos, sirve de escenario imaginario para este «DUELO A MUERTE» que ofrecemos en esta ocasión. Como dos perfectos vaqueros, estos dos hombres se enfrentarán cara a cara, en un juego a muerte en el que sólo uno podrá contarlo.

Es este un juego rápido y de reflejos que consiste, fundamentalmente, en «atinar» al contrario con un disparo certero, y ocasionarle la muerte. Para ello, se puede disparar hacia arriba y hacia abajo, existiendo un ángulo aleatorio de disparo con lo que se consigue un poco más de dificultad en el juego.

Como nota a destacar, la necesidad de acertar al contrario en pleno pecho, de lo contrario, no acabaremos con él. Para lograr nuestro propósito, contamos con diez vidas, cada una de las cuales iremos perdiendo a medida que el contrario nos dispare.

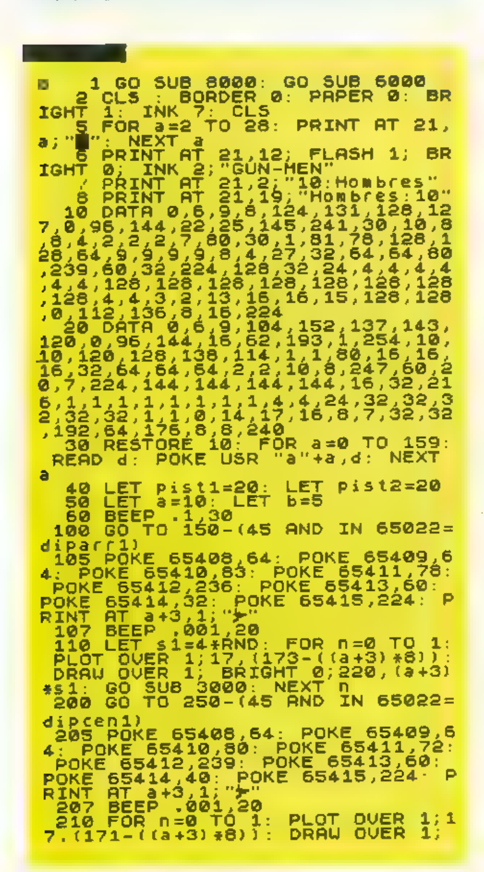
Animo y acción. No se deje intimidar por un simple vaquero.



Coja las pistolas y sea certero. El precio es la muerte. (dib. 1).

NOTAS GRAFICAS

A B C D E F G H I J K L M N O P Q R S T



```
   1 GO SUB 8000: GO SUB 6000
   2 CLS : BORDER 0: PAPER 0: BRIGHT 1: INK 7: CLS
   5 FOR a=2 TO 28: PRINT AT 21,a;"█": NEXT a
   6 PRINT AT 21,12; FLASH 1; BRIGHT 0; INK 2;"GUN-MEN"
   7 PRINT AT 21,2;"10:Hombres"
   8 PRINT AT 21,19;"Hombres:10"
  10 DATA 0,6,9,8,124,131,128,127,0,96,144,22,25,145,241,30,10,8,8,4,2,2,2,7,80,30,1,81,78,128,128,64,9,9,9,9,8,4,27,32,64,64,80,239,60,32,224,128,32,24,4,4,4,4,4,4,128,128,128,128,128,128,128,128,4,4,3,2,13,16,16,15,128,128,0,112,136,8,16,224
  20 DATA 0,6,9,104,152,137,143,120,0,96,144,16,62,193,1,254,10,10,120,128,138,114,1,1,80,16,16,16,32,64,64,64,2,2,10,8,247,60,20,7,224,144,144,144,144,16,32,216,1,1,1,1,1,1,1,1,4,4,24,32,32,32,32,32,1,1,0,14,17,16,8,7,32,32,192,64,176,8,8,240
  30 RESTORE 10: FOR a=0 TO 159: READ d: POKE USR "a"+a,d: NEXT a
  40 LET pist1=20: LET pist2=20
  50 LET a=10: LET b=5
  60 BEEP .1,30
 100 GO TO 150-(45 AND IN 65022=diparr1)
 105 POKE 65408,64: POKE 65409,64: POKE 65410,83: POKE 65411,78: POKE 65412,236: POKE 65413,60: POKE 65414,32: POKE 65415,224: PRINT AT a+3,1;""
 107 BEEP .001,20
 110 LET s1=4*RND: FOR n=0 TO 1: PLOT OVER 1;17,(173-((a+3)*8)): DRAW OVER 1; BRIGHT 0;220,(a+3)*s1: GO SUB 3000: NEXT n
 200 GO TO 250-(45 AND IN 65022=dipcen1)
 205 POKE 65408,64: POKE 65409,64: POKE 65410,80: POKE 65411,72: POKE 65412,239: POKE 65413,60: POKE 65414,40: POKE 65415,224: PRINT AT a+3,1;""
 207 BEEP .001,20
 210 FOR n=0 TO 1: PLOT OVER 1;17.(171-((a+3)*8)): DRAW OVER 1; BRIGHT 0;220,0: GO SUB 3000: NEXT n
 300 GO TO 350-(45 AND IN 65022=dipabj1)
 305 POKE 65408,64: POKE 65409,64: POKE 65410,64: POKE 65411,66: POKE 65412,238: POKE 65413,62: POKE 65414,38: POKE 65415,227: PRINT AT a+3,1;""
 307 BEEP .001,20
 310 LET s1=4*RND: FOR n=0 TO 1: PLOT OVER 1;17,(169-((a+3)*8)): DRAW OVER 1; BRIGHT 0;220,-(22-(a+3))*s1: GO SUB 3000: NEXT n
 400 GO TO 450-(45 AND IN 49150=diparr2)
 405 POKE 65480,2: POKE 65481,2: POKE 65482,202: POKE 65483,114: POKE 65484,55: POKE 65485,60: POKE 65486,4: POKE 65487,7: PRINT AT b+3,29;""
 407 BEEP .001,20
 410 LET s2=4*RND: FOR n=0 TO 1: PLOT OVER 1;230,(173-((b+3)*8)): DRAW OVER 1; BRIGHT 0;-220,(b+3)*s2: GO SUB 3100: NEXT n
 500 GO TO 550-(45 AND IN 49150=dipcen2)
 505 POKE 65480,2: POKE 65481,2: POKE 65482,10: POKE 65483,8: POKE 65484,247: POKE 65485,60: POKE 65486,20: POKE 65487,7: PRINT AT b+3,29;""
 507 BEEP .001,20
 510 FOR n=0 TO 1: PLOT OVER 1;230,(171-((b+3)*8)): DRAW OVER 1; BRIGHT 0;-220,0: GO SUB 3100: NEXT n
 600 GO TO 650-(45 AND IN 49150=dipabj2)
 605 POKE 65480,2: POKE 65481,2: POKE 65482,2: POKE 65483,66: POKE 65484,119: POKE 65485,124: POKE 65486,100: POKE 65487,199: PRINT AT b+3,29;""
 607 BEEP .001,20
 610 LET s2=4*RND: FOR n=0 TO 1: PLOT OVER 1;230,(168-((b+3)*8)): DRAW OVER 1; BRIGHT 0;-220,-(22-(b+3))*s2: GO SUB 3100: NEXT n
2000 LET a=a+(1 AND IN 64510=baj1 AND a<15)-(1 AND IN 63486=sub1 AND a>0)
2010 PRINT AT a,0;"  ";AT a+1,0;"";AT a+2,0;"";AT a+3,0;"";AT a+4,0;"";AT a+5,0;"";AT a+6,0;"  "
2100 LET b=b-(1 AND IN 61438=sub2 AND b>0)+(1 AND IN 57342=baj2 AND b<15)
2110 PRINT AT b,29;"  ";AT b+1,29;"";AT b+2,29;"";AT b+3,29;"";AT b+4,29;"";AT b+5,29;"";AT b+6,29;"  "
2500 GO TO 100
3000 IF ATTR (b+3,29)=7 THEN GO SUB 4000
3010 RETURN
3100 IF ATTR (a+3,1)=7 THEN GO SUB 4100
3110 RETURN
4000 LET pist2=pist2-1
4001 PRINT AT 21,27;"  ";AT 21,27;INT (pist2/2)
4005 BEEP .5,10
4010 IF pist2<1 THEN GO TO 5030
4099 RETURN
4100 LET pist1=pist1-1
4101 PRINT AT 21,2;"  ";AT 21,2;INT (pist1/2)
4105 BEEP .5,10
4110 IF pist1<1 THEN GO TO 5000
4199 RETURN
5000 PAPER 2: CLS
5010 PRINT AT 2,0;"HA PERDIDO EL JUGADOR IZQUIERDO"
5011 PAUSE 100
5020 GO TO 5050
5030 PAPER 5: CLS
5040 PRINT AT 5,0;"HA PERDIDO EL JUGADOR DERECHO"
5050 PAUSE 100
5060 PAPER 4: CLS
5070 INK 0: PRINT AT 10,13;"TURURU"
5080 INPUT "Si quieres jugar de nuevo pulsa          S y ENTER";a$
5090 IF a$="s" THEN GO TO 2: GO TO 5080
6000 REM Eleccion de teclado
6100 GO SUB 9900
6110 PRINT AT 20,0;"Jugador Izquierdo elige tu teclade subida entre el 1 y el 5": BEEP .5,1
6120 PAUSE 0: LET sub1=IN 63486: PRINT AT 5,0; FLASH 1;INKEY$: BEEP .5,20
6210 PRINT AT 20,0;"Jugador Izquierdo elige tu teclade bajada entre la Q y la T"
6220 PAUSE 0: LET baj1=IN 64510: PRINT AT 9,0; FLASH 1;INKEY$: BEEP .5,20
6310 PRINT AT 20,0;"Jugador Izquierdo elige tu teclade disparo hacia arriba a/g"
6320 PAUSE 0: LET diparr1=IN 65022: PRINT AT 6,2; FLASH 1;INKEY$: BEEP .5,20
6410 PRINT AT 20,0;"Jugador Izquierdo elige tu teclade disparo hacia el frente a/g"
6420 PAUSE 0: LET dipcen1=IN 65022: PRINT AT 7,2; FLASH 1;INKEY$: BEEP .5,20
6510 PRINT AT 20,0;"Jugador Izquierdo elige tu teclade disparo hacia abajo a/g        "
6520 PAUSE 0: LET dipabj1=IN 65022: PRINT AT 8,2; FLASH 1;INKEY$: BEEP .5,20
6610 PRINT AT 20,0;"Jugador derecho elige tu tecla  de subida entre el 0 y el 6."
6620 PAUSE 0: LET sub2=IN 61438: PRINT AT 5,30; FLASH 1;INKEY$: BEEP .5,10
6710 PRINT AT 20,0;"Jugador derecho elige tu tecla  de bajada entre la P yla Y.    "
6720 PAUSE 0: LET baj2=IN 57342: PRINT AT 9,30; FLASH 1;INKEY$: BEEP .5,20
6810 PRINT AT 20,0;"Jugador derecho elige tu tecla  de disparo arriba  L/H.    "
6820 PAUSE 0: LET diparr2=IN 49150: PRINT AT 6,28; FLASH 1;INKEY$: BEEP .5,20
6830 PRINT AT 20,0;"Jugador derecho elige tu tecla  de disparo al frente  L/H.    "
6840 PAUSE 0: LET dipcen2=IN 49150: PRINT AT 7,28; FLASH 1;INKEY$: BEEP .5,20
6850 PRINT AT 20,0;"Jugador derecho elige tu tecla  de disparo abajo  L/H.        "
6860 PAUSE 0: LET dipabj2=IN 49150: PRINT AT 8,28; FLASH 1;INKEY$: BEEP .5,20
6900 PRINT #0; FLASH 1;"PULSA UNA TECLA"
6910 PAUSE 0: RETURN
8000 GO SUB 9900
8010 PRINT AT 5.6;"* * * GUN-MEN * * *";AT 15,0;" QUITA EL INTERFACE DE IMPRESORA,SI ES QUE LA TIENES PUESTA.";AT 10,9; FLASH 1;"INSTRUCCIONES": GO SUB 9990
8020 GO SUB 9900: PRINT AT 5,0,"   ESTE JUEGO HA DE SER JUGADO ENTRE DOS ASI QUE SI ESTAS SOLO BUSCATE UN AMIGO,SI YA LO TIENES PUEDES CONTINUAR": GO SUB 9990
8030 GO SUB 9900: PRINT AT 5,0;"  COMO SUPONEIS SE TRATA DE DISPARAR SIN PIEDAD,PERO CON UNA CONDICION HAY QUE DAR EN EL CORAZON,CUALQUIER OTRA ZONA DEL CUERPO NO SE TIENE EN CUENTA.";AT 15,0;"  AHORA TENEIS QUE ELEGIR LAS TECLAS DE MANDO !!!SUERTE!!!"
8040 GO SUB 9990: RETURN
8990 PAUSE 0
9900 CLS : BEEP .5,10: BORDER 4: PAPER 4: INK 0: CLS : PRINT AT 1,3;"DIBUJOS: JULIA PEREZ SERRANO";AT 0,2;"© ESTHER VIEJO,MANUEL ORCERA";AT 2,10;"GRUPO MOVE": RETURN
9990 PRINT #0;"Pulsa una tecla": PAUSE 0: BEEP .1,0: RETURN
```

LA AGENDA ELECTRONICA

Emilio OTERMIN, Madrid

Premiado con 15.000 Ptas.

Spectrum 48 K

Sea práctico y convierta a su ordenador en una espléndida y completa agenda. Este es el objetivo del programa que ofrecemos a continuación y que, sin lugar a dudas, le será de una enorme utilidad.

Este programa permite la gestión de un fichero de datos o agenda. El método de identificación de nombres escogido, permite que aparezcan en pantalla todos los items que contengan la cadena de búsqueda elegida, comenzando por el primer carácter. La variable «a» de la línea 5, posibilita ajustar la dimensión de las matricea h$ para nombre, ¡$ dirección y j$ teléfono. Siguiendo las instrucciones que aparecen en pantalla, no se debe encontrar ningún problema. El valor de la variable «a», que es la que determina la dimensión de las matrices de almacenamiento, está ajustada por el autor al valor de 35, pero puede aumentarse a 300 como mínimo. Para adecuar este programa al SPECTRUM de 16 K, deben eliminarse todas las sentencias que manejan el MICRODRIVE y ajustar el valor de la susodicha variable «a». Una vez introducido el listado, hacer RUN: el programa preguntará si queremos instrucciones, luego aparecerá la palabra CREACION al ser la primera vez que se ejecuta; en la opción 4 del MENU PRINCIPAL (versión cassette), el programa se autograbará junto con los datos y al ser cargado la próxima vez, se autoejecutará desde el MENU PRINCIPAL. Atención, ESTO NO OCURRIRA SI EMPLEAMOS MICRODRIVE para grabar datos; con el drive se graban las matrices de nombres, direcciones y teléfonos bajo el nombre que escojamos de FORMA SEPARADA, así que, a la hora de volverlas a cargar, habrá que hacer LOAD 3 veces.

```
   1 REM fichero
   2 REM © EMILIO OTERMIN
   5 LET a=35: LET cr=210: LET e=0: LET f=0: LET g=0: LET imp=0: LET a$="Pulsa Nro. opcion elegida": LET b$="Pulsa una tecla"
   7 LET c$="Nombre ? ": LET d$="Direccion ? ": LET e$="Telefono ? ": LET f$="Muy largo, maxima longitud ": LET l$="MENU PRINCIPAL": LET m$="
"
  10 POKE 23609,20: BORDER 4: PAPER 4: PRINT AT 10,8; INK 7; PAPER 2;" F I C H E R O ";AT 20,1; INK 0; PAPER 6;"Quieres instrucciones ? (s/n)": IF INKEY$="s" THEN LET in=30: GO TO 5300
  20 IF INKEY$<>"n" THEN GO TO 10
  30 CLS : GO TO cr
  80 CLS : PRINT AT 2,8; INK 7;
```

Si le echa un poco de imaginación, verá a su Spectrum como a una bella secretaria. (dib. 2).

A. PEREDA

```
PAPER 2;" ";l$;" ";AT 6,9; INK 0
; PAPER 4;"1- CONSULTA";AT 9,9;"
2- MODIFICACION";AT 12,9;"3- ESP
ACIO LIBRE";AT 15,9;"4- SALIDA";
AT 18,9;"5- INSTRUCCIONES"
  85 PRINT AT 21,4; PAPER 5;a$
  90 LET g=53: GO SUB 8500
 100 LET in=80: GO TO b*100
 210 DIM h$(a,25): DIM i$(a,25):
 DIM j$(a,10): LET c=0
 220 CLS : LET c=c+1: IF c=a+1 T
HEN PRINT AT 10,7; PAPER 1; INK
7;"FICHERO  COMPLETO";AT 16,8; P
APER 5; INK 0;b$: PAUSE 0: IF e=
1 THEN LET e=0: GO TO 5031
 225 IF c=a+1 THEN GO TO 80
 227 IF e<>1 THEN PRINT AT 2,11;
 INK 7; PAPER 2;" CREACION "
 230 PRINT AT 5,9; PAPER 1; INK
7;" STOP para salir"
 232 INPUT (c$);g$: IF LEN g$>25
 THEN PRINT AT 12,1; PAPER 1; IN
K 7;f$;25: GO TO 232
 235 IF CODE g$=0 OR CODE g$=32
THEN GO TO 232
 240 IF CODE g$=226 THEN LET f=1
: IF e=1 THEN LET e=0: GO TO 503
1
 245 IF f=1 THEN LET f=0: GO TO
80
 250 LET mod=1: LET h$(c)=g$: PR
INT m$;AT 20,0;c$;g$
 252 PRINT AT 5,8;m$
 253 INPUT (d$);g$: IF LEN g$>25
 THEN PRINT AT 12,1; PAPER 1; IN
K 7;f$;25: GO TO 253
 254 IF CODE g$=0 OR CODE g$=32
THEN GO TO 253
 255 LET i$(c)=g$: PRINT m$;AT 2
1,0;d$;g$
 257 INPUT (e$);g$: IF LEN g$>10
 THEN PRINT AT 12,1; PAPER 1; IN
K 7;f$;10: GO TO 257
 258 IF CODE g$=0 OR CODE g$=32
THEN GO TO 257
 260 LET j$(c)=g$: GO TO 220
4900 CLS : PRINT AT 2,7; PAPER 2
; INK 7;" MENU DE CONSULTA ";AT
6,8; PAPER 4; INK 0;"1- Por ";c$
;AT 9,8;"2- Por ";d$;AT 12,8;"3-
 Por ";e$;AT 15,8;"4- Listado Ge
neral";AT 18,8;"5- ";l$;AT 21,4;
 PAPER 5;a$
4910 LET g=53: GO SUB 8500
4920 LET b=b-48: GO TO b*10+4920
4930 INPUT (c$);k$: IF LEN k$>25
 THEN PRINT AT 12,1;f$;25: GO TO
 4930
4931 IF CODE k$=0 OR CODE k$=32
THEN GO TO 4930
4932 LET d=0: GO SUB 6500: GO SU
B 9500: FOR b=1 TO c
4933 IF h$(b)(1 TO LEN k$)=k$ TH
EN GO SUB 9000: LET d=1: IF g=1
THEN GO SUB 5041
4934 IF g=1 THEN GO SUB 6500
4935 NEXT b
4936 IF d<>1 THEN GO SUB 8000
4937 IF g=1 THEN LET g=0: RETURN

4938 GO TO 4900
4940 INPUT (d$);k$: IF LEN k$>25
 THEN PRINT AT 12,1;f$;25: GO TO
 4940
4941 IF CODE k$=0 OR CODE k$=32
THEN GO TO 4940
4942 LET d=0: GO SUB 6500: GO SU
B 9500: FOR b=1 TO c
4943 IF i$(b)(1 TO LEN k$)=k$ TH
EN GO SUB 9000: LET d=1
4944 NEXT b
4945 IF d<>1 THEN GO SUB 8000
4946 GO TO 4900
4950 INPUT (e$);k$: IF LEN k$>10
 THEN PRINT AT 12,1;f$;10: GO TO
 4950
4951 IF CODE k$=0 OR CODE k$=32
THEN GO TO 4950
4952 LET d=0: GO SUB 6500: GO SU
B 9500: FOR b=1 TO c
4953 IF j$(b)(1 TO LEN k$)=k$ TH
EN GO SUB 9000: LET d=1
4954 NEXT b
4955 IF d<>1 THEN GO SUB 8000
4956 GO TO 4900
4960 GO SUB 6500: GO SUB 9500
4961 LET imp=2: PRINT AT 10,6; P
APER 6;"Por impresora ? (s/n)":
IF INKEY$="s" THEN LET imp=1
4962 IF INKEY$<>"n" THEN GO TO 4
961
4963 CLS : FOR b=1 TO c
4964 GO SUB 9010
4965 NEXT b: LET imp=0
4966 PRINT TAB 8; PAPER 5;b$: PA
USE 0: GO TO 4900
4970 GO TO 80
5000 CLS : PRINT AT 5,5; PAPER 2
; INK 7;" MENU DE MODIFICACION "
; PAPER 4; INK 0;AT 10,8;"1- Amp
liar";AT 12,8;"2- Suprimir";AT 1
4,8;"3- ";l$;AT 21,4; PAPER 5;a$
5010 LET g=51: GO SUB 8500
5020 LET b=b-48: GO TO b*10+5020
5030 GO SUB 6500: GO SUB 9500: L
ET e=1: GO TO 220
5031 GO TO 5000
5040 CLS : LET g=1: GO SUB 4930:
 GO TO 5000
5041 PRINT AT 16,2; INK 0; PAPER
 6;"Suprimo del fichero ? (s/n)"
5042 IF INKEY$="s" THEN LET mod=
1: GO SUB 9500: LET h$(b)=h$(c):
 LET i$(b)=i$(c): LET j$(b)=j$(c
): LET h$(c)="": LET i$(c)="": L
ET j$(c)="": GO TO 5044
5[illegible] : IF INKEY$<>"n" THEN GO TO 5
042
5044 RETURN
5050 GO TO 80
5100 CLS : GO SUB 6500: CLS : GO
 SUB 9500: PRINT AT 10,11; PAPER
 7;a-c;" nombres": IF a-c=1 THEN
 PRINT AT 10,19; PAPER 4;" "
5110 PRINT AT 18,9; PAPER 5;b$:
PAUSE 0: GO TO 80
5200 CLS : IF mod<>1 THEN POKE ■
3609,0: STOP
5205 PRINT AT 4,8; INK 7; PAPER
2;" MENU DE SALIDA ";AT 10,5; IN
K 0; PAPER 4;"1- GRABAR EN CASSE
TTE";AT 12,5;"2- GRABAR EN MICRO
DRIVE";AT 14,5;"3- ";l$;AT 20,4;
 PAPER 5;a$
5210 LET g=51: GO SUB 8500
5215 LET b=b-48: IF b=1 THEN GO
TO 5220
5216 IF b=2 THEN GO TO 5250
5217 GO TO 80
5220 CLS : LET mod=0: LET cr=80:
 PRINT AT 9,6; PAPER 6;" ";b$;"
 ";AT 10,6;m$(1 TO 19);AT 11,6;
"para grabar fichero": PAUSE 0
5225 CLS : PRINT AT 10,8; PAPER
1; INK 7; FLASH 1;" G R A B A N
D O ": SAVE "fichero" LINE 10
5230 PRINT AT 6,4; PAPER 6;"Para
 verificar grabacion";AT 7,4;m$(
1 TO 24);AT 8,4;"   prepara cas
sette   ";AT 9,4;m$(1 TO 24);AT
;10,4;"  y despues pulsa PLAY  "
5235 PRINT AT 14,7; PAPER 1; INK
 7;"Si no verifica...";AT 15,7;m
$(1 TO 17);AT 16,7;"   GOTO 522
0   ": VERIFY ""
5240 GO TO 80
5250 CLS : CAT 1: PRINT AT 20,4;
 PAPER 5;b$: PAUSE ■
5253 LET z$="le das a calles,dir
e-cciones y telefonos ? ": GO SU
B 5260
5254 SAVE *"m";1;w$ DATA h$(): S
AVE *"m";1;x$ DATA i$(): SAVE *"
m";1;y$ DATA j$(): GO TO 5270
5256 IF INKEY$<>"n" THEN GO TO 5
255
5257 GO TO 5250
5260 CLS : INPUT "Que nombre ";(
z$);w$,x$,y$: IF LEN w$>10 OR LE
N x$>10 OR LEN y$>10 THEN PRINT
AT 10,1; PAPER 1; INK 7;f$;10;AT
 11,8; PAPER 5; INK 0;b$: PAUSE
0: GO TO 5260
5261 RETURN
5270 PRINT AT 9,13; PAPER 5;"GRA
BADO";AT 10,9;b$: PAUSE 0: GO TO
 80
5300 CLS : PRINT AT 2,9; INK 7;
PAPER 2;" INSTRUCCIONES ";AT 6,4
; INK 0; PAPER 4;"1- Es autoexpl
icativo";AT 8,4;"2- Nombres y di
recciones,";AT 9,7;"maximo 25 ca
racteres y";AT 10,7;"en telefono
s, 10"
5310 PRINT AT 12,4;"3- STOP se i
ntroduce con";AT 13,7;"SIMBOL SH
IFT y A";AT 15,4;"4- Si hubo cam
bios, se";AT 16,7;"autograbara a
l elegir";AT 17,7;"opcion 4 en e
l";AT 18,7;l$
5320 PRINT AT 21,9; PAPER 5,b$:
PAUSE 0: CLS : GO TO in
6500 CLS : PRINT AT 10,8; PAPER
1; INK 7;" B U S C A N D O ": PA
USE 20: RETURN
8000 PRINT AT 10,6; PAPER 1; INK
 7;"No figura en fichero";AT 14,
9; PAPER 5; INK 0;b$: PAUSE ■
8010 RETURN
8500 PAUSE 0: LET b=CODE INKEY$:
 IF b<49 OR b>g THEN GO TO 850■
8510 CLS : RETURN
9000 IF imp=1 THEN LET imp=0: LP
RINT h$(b)'i$(b)'j$(b)'': RETURN
9005 CLS
9010 PRINT h$(b)'i$(b)'j$(b)'':
IF imp=2 THEN RETURN
9020 IF g<>1 THEN PRINT AT 12,3;
 PAPER 6;"Quieres imprimirlo ? (
s/n)": IF INKEY$="s" THEN LPRINT
 h$(b)'i$(b)'j$(b)'': RETURN
9025 IF g=1 THEN RETURN
9030 IF INKEY$<>"n" THEN GO TO 9
020
9040 GO SUB 6500: RETURN
9500 LET c=0: FOR f=1 TO a
9510 IF CODE h$(f)=32 THEN LET c
=f-1: LET z=f: LET f=a
9520 NEXT f: IF z=1 THEN GO TO 9
530
9525 IF c=0 THEN LET c=a
9530 RETURN
```

SOLITARIO

Tomás FERNANDEZ MARTINEZ, Madrid

Premiado con 15.000 Ptas.

Spectrum 16 K

Para los amantes de las cartas, los solitarios suelen ser buenos recursos para pasar un rato agradable en el calor del hogar. Pues bien, con su ordenador puede también conseguir esos placenteros momentos de ocio siguiendo, tan sólo, las instrucciones que a continuación les ofrecemos.

El «SOLITARIO» consiste en llevar a cabo en la pantalla del ordenador, el mismo procedimiento utilizado en los naipes, es decir, una vez colocadas todas las cartas boca abajo, conseguir situarlas ordenadamente antes de que aparezcan los cuatro reyes de sus correspondientes palos.

Para su realización, el Spectrum se encarga de barajar las cartas aleatoriamente, igual que si un jugador lo hiciera con los naipes y, una vez ordenadas boca abajo, veremos en la parte inferior de la pantalla, una carta que se situará en su lugar correspondiente de la siguiente manera: la fila de arriba corresponde a los oros; la segunda, a las copas; la tercera, a las espadas, y la cuarta, a los bastos; además, las cartas se situarán, dentro de cada fila, en orden ascendente, de izquierda a derecha.

Si conseguimos colocar todas las cartas en su sitio adecuado antes de la aparición de los cuatro reyes, habremos ganado el juego, lo mismo que si colocamos los reyes cuando todos los naipes se encuentren debidamente colocados.



Con su Spectrum, hacer solitarios será más divertido.

NOTAS GRAFICAS

A B C D

```
   2 INK 0: PAPER 6: BORDER 1
   4 DIM m(40,5): LET x2=0
   8 GO SUB 9500
  10 PRINT AT 4,5; INVERSE 1;"■■
■■ SOLITARIO ■■■■"
  20 PRINT AT 7,4;"El solitario
consiste en   poner todas las c
artas en su   lugar antes de qu
e salgan los  cuatro reyes .
              Si cuando hay
an salido los cuatro reyes las
demas cartas  estan en su sitio
 tambien ganas."
  30 PRINT AT 17,4;"! Suerte !"
  40 PRINT AT 19,1;"Pulse una te
cla para seguir"
  50 PAUSE 0
  60 CLS
 110 FOR y=168 TO 72 STEP -32
 112 FOR x=7 TO 223 STEP 24
 120 PLOT x,y: DRAW 17,0: DRAW 0
,-25: DRAW -17,0: DRAW 0,25
 128 LET x2=x2+1
 130 LET y1=y/8: LET x1=INT ((x/
8)+1)
 132 LET m(x2,1)=x1
 134 LET m(x2,2)=22-y1
 135 INK 4
 140 FOR a=0 TO 1
 150 FOR b=1 TO 3
 160 PRINT AT 21-y1+b,x1+a;"■"
 170 NEXT b: NEXT a
 172 INK 0
 180 NEXT x: NEXT y
 182 LET x2=0
 184 PRINT AT 18,6;"Espere,por f
avor"
 190 FOR a=1 TO 10
 200 FOR b=1 TO 4
 210 LET c=INT ((RND*40)+1)
 220 IF m(c,4)<>0 THEN GO TO 210
 230 LET m(c,3)=a: LET m(c,4)=b
 240 NEXT b: NEXT a
 242 PRINT AT 18,6;"
    "
 250 PLOT 103,32: DRAW 17,0: DRA
W 0,-25: DRAW -17,0: DRAW 0,25
 500 FOR a=1 TO 40
 510 LET z=m(a,4)*10+m(a,3)-10
 530 IF m(a,3)=9 THEN LET m(a,3)
=11
 540 IF m(a,3)=10 THEN LET m(a,3
)=12: LET x2=x2+1
 542 IF m(a,3)=8 THEN LET m(a,3)
=10
 546 GO SUB 8000: PRINT AT 18,13
;"  "
 550 PRINT AT 18,13;m(a,3);AT 19
,13;w$;AT 20,14;w$
 556 PRINT AT 21,0;"Pulse tecla"
 560 PAUSE ■
 562 FOR f=0 TO 1: FOR b=0 TO 2
 564 PRINT AT m(z,2)+b,m(z,1)+f;
"  "
 566 NEXT b: NEXT f
 568 LET m(z,5)=1
 570 PRINT AT m(z,2),m(z,1);m(a,
3);AT m(z,2)+1,m(z,1);w$;AT m(z,
2)+2,m(z,1)+1;w$
 572 LET k=a+1
 574 IF x2=4 THEN GO TO 590
 580 NEXT a
 590 LET a=k
 600 FOR r=1 TO 40
 610 IF m(r,5)=1 THEN GO TO 710
 620 GO SUB 8000
 624 IF m(a,3)=10 THEN LET m(a,3
)=12
 626 IF m(a,3)=9 THEN LET m(a,3)
=11
 628 IF m(a,3)=8 THEN LET m(a,3)
=10
 630 LET z=m(a,4)*10+m(a,3)-10
 640 PRINT AT m(r,2),m(r,1);m(a,
3);AT m(r,2)+1,m(r,1);w$;AT m(r,
2)+2,m(r,1)+1;w$
 700 LET a=a+1
 702 PRINT AT 21,0;"
    "
 710 NEXT r
 714 PRINT AT 18,0;"Han salido l
os cuatro reyes.    Si el resto
de las cartas estan en su lugar
habra ganado."
 716 INPUT "Desea volver a jugar
? s/n  ";w$
 718 CLS
 720 IF w$="s" THEN RUN
 730 STOP
8000 IF m(a,4)=1 THEN LET w$="■"
8010 IF m(a,4)=2 THEN LET w$="■"
8020 IF m(a,4)=3 THEN LET w$="■"
8030 IF m(a,4)=4 THEN LET w$="■"
8040 RETURN
9500 FOR x=144 TO 147
9510 FOR c=0 TO 7
9520 READ w
9530 POKE USR CHR$ x+c,w
9540 NEXT c: NEXT x
9550 DATA 0,24,60,126,126,60,24,
0
9560 DATA 126,126,126,126,60,24,
24,60
9570 DATA 16,16,16,16,16,16,124,
16
9580 DATA 24,28,60,60,60,56,24,1
6
9590 RETURN
```

El color en el Spectrum (y 4)

UTILIZANDO LA MEMORIA

Gabriel NIETO

En este último capítulo sobre EL COLOR EN EL SPECTRUM, vamos a analizar un punto muy importante. Una vez conocida la distribución de atributos del ordenador, hay que saber cuál es el contenido de cada dirección de memoria y cómo podemos modificar éste, según sean las necesidades de programación.

Con lo visto en el capítulo anterior, ya sabemos cómo está distribuida la pantalla, pero ¿cuál es el contenido de cada dirección de memoria y cómo podemos modificarlo?

Si desconectamos nuestro ordenador y seguidamente lo volvemos a conectar, habremos borrado toda la información que permanecía en la memoria RAM, por lo que partiremos con los valores iniciales, situados allí por el fabricante, de cada celdilla de memoria. ¿Qué quiere decir esto? Tecleemos la siguiente línea:

1Ø PRINT PEEK 23ØØØ

Manejo de la memoria

En la pantalla nos aparecerá el número 56. ¿Por qué este número? Recordemos que el comando PEEK nos sirve para explorar las direcciones de memoria y ver el contenido de cada una de ellas. De forma que el número 56, en este caso, es el contenido de la celdilla 23ØØØ.

Para entender esto, tenemos que recurrir a la composición de los números binarios. El número 56 en el sistema binario sería:

Ø Ø 1 1 1 Ø Ø Ø

Si tenemos que tener en cuenta el valor de cada posición del número binario, se quedaría de izquierda a derecha como sigue:

128	64	32	16	8	4	2	1
Ø	Ø	1	1	1	Ø	Ø	Ø

Los números subrayados representan el valor que tiene cada posición de abajo cuando ésta se encuentra activada. Para que esto ocurra, el bit correspondiente tiene que tener el valor 1. En el ejemplo anterior, los bits activados tienen los valores 32, 16 y 8, la suma de los cuales nos da el número 56, que era el valor que tenía la dirección de memoria 23ØØØ. Las ocho partes del número binario forman un octeto, ya que están compuestas por ocho bits; cada uno de los cuales, como ya hemos dicho, puede estar activado o no. Si lo está, tendrá el valor 1, y si no, tendrá el valor Ø. Cuando se encuentran activados, tendrán los valores correspondientes a la suma que les corresponda de cada lugar de la fila de arriba (128, 64, ... 2, 1). Así, el número máximo que podemos obtener de este modo es el 1 1 1 1 1 1 1 1, que será el 255, ya que la suma de sus bits activados nos da dicho valor.

En los octetos se introduce toda la información sobre el color.

Hay que tener en cuenta que en los números binarios se empieza a contar de derecha a izquierda, por tanto, el esquema del octeto sería el siguiente:

	128	64	32	16	8	4	2	1
BITS	7.º	6.º	5.º	4.º	3.º	2.º	1.	Ø.º

Esto tiene su lógica, porque 128, por ejemplo, es el resultado de elevar 2 a la 7.ª potencia; 64 de elevarlo a la 6.ª, y así con todas las demás posiciones.

Teniendo en cuenta esto último, el carácter tendrá el valor:

128— Si está activado el FLASH.

64— Si está activado BRIGHT.

8— Multiplicado por el código de color del papel (de Ø a 7).

El valor del código del color de tinta (de Ø a 7), que se forma combinando los valores de los tres primeros bits.

Por ejemplo, un carácter impreso con FLASH, papel azul y tinta blanca, está representado por el valor 143:

128 por estar activado el FLASH

8 por el color del papel, que se obtiene multiplicando el código del papel azul (que es 1) por ocho.

7 correspondiente al código de la tinta blanca.

La suma de todos ellos nos da 143. Haga la prueba con diferentes tipos de caracteres.

A continuación vamos a introducir el siguiente programa, que nos va a dar el contenido de todas las direcciones de memoria que componen el fichero de atributos:

```
10 FOR I=22528 TO 23295
20 PRINT I,TAB 8;PEEK I
30 NEXT I
```

En todas las posiciones obtenemos el valor 56, como ocurría anteriormente en la posición 23ØØØ, que correspondía al carácter de la fila 14, columna 24. Este valor se obtiene multiplicando por 8 el color del papel (blanco cuando conectamos el ordenador). Como toda la pantalla está en blanco, todas las direcciones de la memoria del archivo de atributos tendrán el mismo valor.

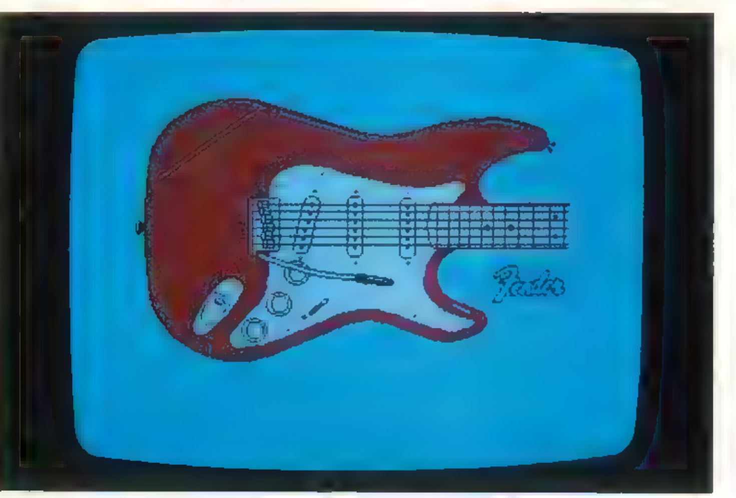
Sin embargo, vamos a probar a introducir una nueva línea en el programa anterior:

```
  5 PRINT AT 10,0; PAPER 1; INK
6;" "
```

Al llegar a la dirección 22538, veremos que aparece el número 14 en lugar del 56. Esto ocurre porque hemos modificado dicho carácter en la línea 5. Ahora al multiplicar 8 por el código del color azul y sumarle el de la tinta, nos da 14 (8 × 1 + 6).

La utilidad del POKE

Otra forma de introducir directamente los códigos de color en el ordenador es utilizando la sentencia POKE. Esta tiene la peculiaridad de que puede cambiar el valor almacenado en una dirección de memoria, sustituyéndole por uno nuevo, que va a producir un efecto diferente en la pantalla. Así, por ejemplo, POKE 23ØØØ, 248 nos muestra un cuadrado en la fila 14, columna 24, con tinta negra, papel blanco, brillo y flash activado. Sería lo mismo que introducir la línea



El código máquina nos permite colorear la pantalla de forma casi inmediata.

```
  10 PRINT AT 14,24; FLASH 1; BR
IGHT 1; PAPER 7; INK 0;" "
```

Pero como podemos ver, el hacerlo con POKE es mucho más directo y práctico al mismo tiempo. Podemos cambiar los valores de la dirección 23ØØØ para observar cada vez un carácter diferente.

Con el programa que les ponemos a continuación, ustedes tienen que introducir los valores de x a través del IMPUT y verán como la pantalla completa toma el color del código que le han introducido. Prueben con varios números para conseguir el efecto deseado, teniendo en cuenta siempre lo que decíamos anteriormente sobre los códigos de color, en relación con FLASH. PAPER. ...

```
10 INPUT X
20 FOR I=22528 TO 23295
30 POKE I,X
40 NEXT I
50 GO TO 10
```

Si ha probado este programa, habrá observado que el ordenador ha coloreado también las líneas 22 y 23, las cuales normalmente no se pueden usar, ya que están reservadas para que su Spectrum emita mensajes o bien para que usted introduzca datos, mediante IMPUT. Sin embargo, al utilizar POKE, podemos tener acceso libre a estas dos líneas, si bien al final de haber realizado nuestras órdenes, el ordenador emitirá un mensaje OK, que nos las borrará de nuevo. Esto podría evitarse utilizando el código máquina.

Podemos también introducir un programa que nos coloree la imagen de forma aleatoria:

```
10 FOR I=22528 TO 23295
20 LET X=RND*255
30 POKE I,X
40 NEXT I
```

Un programa en código máquina

Con todo lo aprendido hasta ahora, hemos empezado a conocer el archivo de atributos. Para sacarle el máximo rendimiento, tendremos que recurrir ineludiblemente al empleo del código máquina. Pero no se preocupen, ya sabemos que son muchos los lectores que tienen cierto respeto a este tema, por eso no vamos a tratar aquí de explicarles tan complicado lenguajes. Ahora interesa mostrarles la rapidez con que dicho léxico actúa en relación al BASIC.

Para ello, vamos a comparar dos programas que hacen exactamente lo mismo, uno de ellos escrito en Basic y el otro en código máquina.

Los dos llenan la pantalla del televisor con tres colores diferentes, a partir de las mismas posiciones. El primero está escrito en Basic

```
  10 FOR I=22528 TO 23295
  20 IF I>=22784 AND I<=23007 TH
EN POKE I,24: GO TO 50
  30 IF I>=23008 AND I<=23295 TH
EN POKE I,16: GO TO 50
  40 POKE I,32
  50 NEXT I
```

Al ejecutarlo, verá cómo la pantalla se va llenando poco a poco con tres colores: verde, magenta y rojo. En este programa hemos utilizado el método directo (a través de POKE) del que hablábamos antes. A pesar de ello, tarda en ejecutarse aproximadamente, alrededor de 14 segundos, tiempo bastante considerable, que no nos permitiría incluirlo como una rutina, ya que su lentitud le restaría bastante interés al programa.

El POKE nos permite introducir directamente los códigos de color.

A continuación vamos a hacer lo mismo, pero esta vez en código máquina. Para ello utilizaremos un programa Basic, donde incluiremos dicho código a través de varias sentencias DATA.

```
  10 FOR I=30000 TO 30036
  20 READ L: POKE I,L
  30 NEXT I
  40 DATA 33,0,88,62,32,119,35,6
2,89,188,32,247
  50 DATA 33,0,89,62,24,119,35,6
2,90,188,32,247
  60 DATA 33,0,90,62,16,119,35,6
2,91,188,32,247,201
  70 RANDOMIZE USR 30000
```

Este programa tarda en ejecutarse apenas un segundo, lo que comparado con el anterior, supone una considerable diferencia. Los números que están escritos en blanco sobre fondo oscuro, son los que contienen los códigos de color. Recordemos que el 32 en el sistema binario era Ø Ø 1 Ø Ø Ø Ø Ø. El bit activado en este caso es el 6.º, al que le correspondía el valor 32. Si quisiéramos, por tanto, cambiar los colores del programa anterior y sustituirlos por otros, sólo tendríamos que buscar los códigos de color que nos interesaran a través del sistema binario, y sustituirlo en forma decimal en la sentencia DATA, por el número escrito en blanco sobre fondo oscuro.

Como ejemplo, vamos a sustituir el 32 por un color verde con el flash activado. Recordemos que el FLASH se activa con el 7.º bit, por tanto éste deberá estar en la posición 1. El color de la tinta se conseguirá activando el 3.º bit.

128	64	32	16	8	4	2	1
1	Ø	Ø	Ø	Ø	1	Ø	Ø

Este número corresponde al 132 decimal. Sustitúyalo y verá cómo cambia el efecto del primer bloque de pantalla.

Utilidad

Este método nos puede servir para colorear la imagen en nuestros programas, de forma automática. Dominar el código máquina en este sentido nos puede resultar muy útil, ya que no podemos olvidar que éste es la base de nuestro ordenador. Todas las instrucciones del Spectrum se traducen desde el Basic, para que el microprocesador las entienda. Es por eso que, al escribirlas en el formato del código original, la velocidad con que se realizan las instrucciones es mucho más rápida. Además, en este formato podemos llegar a lugares que están prohibidos para el lenguaje Basic, sacándole de esta forma mucho más provecho al ordenador.

¡AMPLIA TU MEMORIA A 64 K! (I)

Primitivo de FRANCISCO

Es normal para los que desean iniciarse en el BASIC y se deciden por un Spectrum, hacerlo en favor de la versión 16 K. Esta decisión se toma únicamente por razones económicas; pero pronto resulta fácil percatarse de la necesidad de más memoria.

Afortunadamente la tarjeta de circuito impreso del Spectrum está prevista para las dos versiones, siendo la básica de 16 K bytes; por tanto, su ampliación en 32 K bytes más para conformar los 48 K bytes, resulta fácil; tan sólo es preciso efectuar unos puentes e insertar en los zócalos libres los circuitos integrados para los que están dispuestos. El costo de esta ampliación viene a ser entre ocho y diez mil pesetas y los componentes necesarios se encuentran en el mercado en forma de kit o se pueden comprar de forma suelta.

Para efectuar la ampliación, tan sólo hay que adquirir previamente la información necesaria y decidirse. En este apartado damos las razones de cómo y por qué hacerlo.

64 K Bytes de espacio direccionable

El espacio total de memoria direccionable en el Spectrum es de 64 K bytes (1 K Byte es igual a 1.024 bytes. Se tomó este valor, tan aparentemente extraño, por ser el resultado de potencia binaria más próximo al número 1.000); 64 K bytes son, en realidad, 65536 direcciones, cada una de las cuales se corresponde con un octeto o byte.

Estos 64 K bytes se pueden dividir en cuatro áreas funcionales para el Spectrum, a las cuales llamaremos páginas. Cada una de estas páginas es de 16 K bytes (16384 octetos) y tienen funciones específicas dentro de la totalidad del microordenador. A nivel Hardware el microprocesador se dirige a cada una de ellas con sólo poner el valor binario adecuado en los bits A14 y A15 del bus de direcciones. El resto de los bits, desde AØ hasta A13, son usados para direccionar dentro de cada página.

— Página Ø: Abarca desde la dirección Ø hasta la 16383 (en hexadecimal desde ØØØØH hasta 3FFFH). Se encuentra totalmente ocupada por la ROM en que reside el BASIC; es por tanto inalterable su contenido y en ella están ubicadas las subrutinas a que apunta el microprocesador trás un reset (Dirección ØØØØH), una interrupción enmascarable (Dirección ØØ38H) o una interrupción no enmascarable (Dirección ØØ66H). Para acceder a esta página los bits A14 y A15 son puestos a cero.

— Página 1: Abarca desde la dirección 16384 hasta la 32767 (En hexadecimal desde 4ØØØH hasta 7 FFFH). Esta página se encuentra ocupada por ocho integrados de RAM dinámica, con una disposición interna de 16384 direcciones por un solo bit, por lo que hacen falta ocho bips para conformar el bloque de memoria de 16 K bytes.

Este bloque se encuentra controlado para su refresco por el Z-8Ø y la ULA.

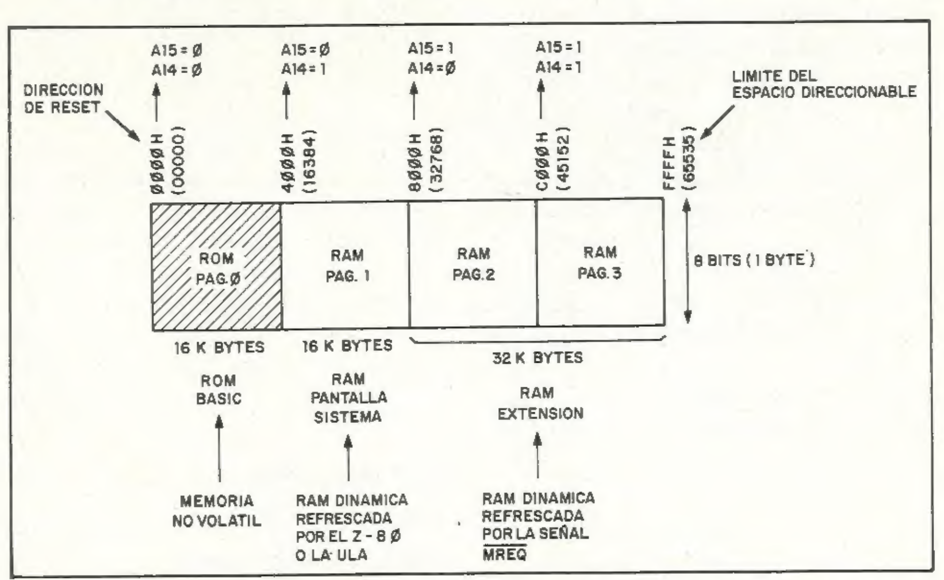
El hecho de que estos dos dispositivos pudieran acceder simultáneamente a las mismas posiciones de memoria, obliga a que uno de los dos tenga prioridad. En nuestro caso, dicha prioridad se le ha dado a la ULA, la cual genera y controla el reloj del sistema enviando hacia el microprocesador una señal de clock de aproximadamente 3,5 MHz. La ULA aprovecha esta circunstancia para ajustar el período del reloj evitando así los tiempos de coincidencia simultánea en el acceso a RAM. Debido a esto, el clock de la CPU no siempre es constante. En este bloque se hallan las principales direcciones del sistema como son: el área de presentación por pantalla y atributos, buffer de impresora, área de variables, etc.

Los dos bits del bus de direcciones que seleccionan esta página, se encuentran en los siguientes estados lógicos: A15 = Ø y A14 = 1.

— Páginas 2 y 3: Abarcan desde las direcciones 32768 hasta la 45151 (8ØØØH y BFFFH) para la página dos, y desde la 45152 hasta la 65535 (CØØØH y FFFF H) para la página tres. Los bits A15 y A14 del bus de direcciones se encuentran a 1 y Ø respectivamente para la página 2 y ambos a 1 para la tres.

Estas dos páginas de 16 K bytes cada una, conforman un bloque de 32 K bytes que, en realidad, se direccionan juntas mediante el bit A15 cuando éste es puesto a 1 por la CPU.

Este bloque es el que se encuentra libre en la versión Spectrum 16 K, y ocupado por memoria RAM DINAMICA en la versión 48 K.



Paginación del espacio de memoria en el Spectrum y sus cotas de dirección. (Fig. 1).

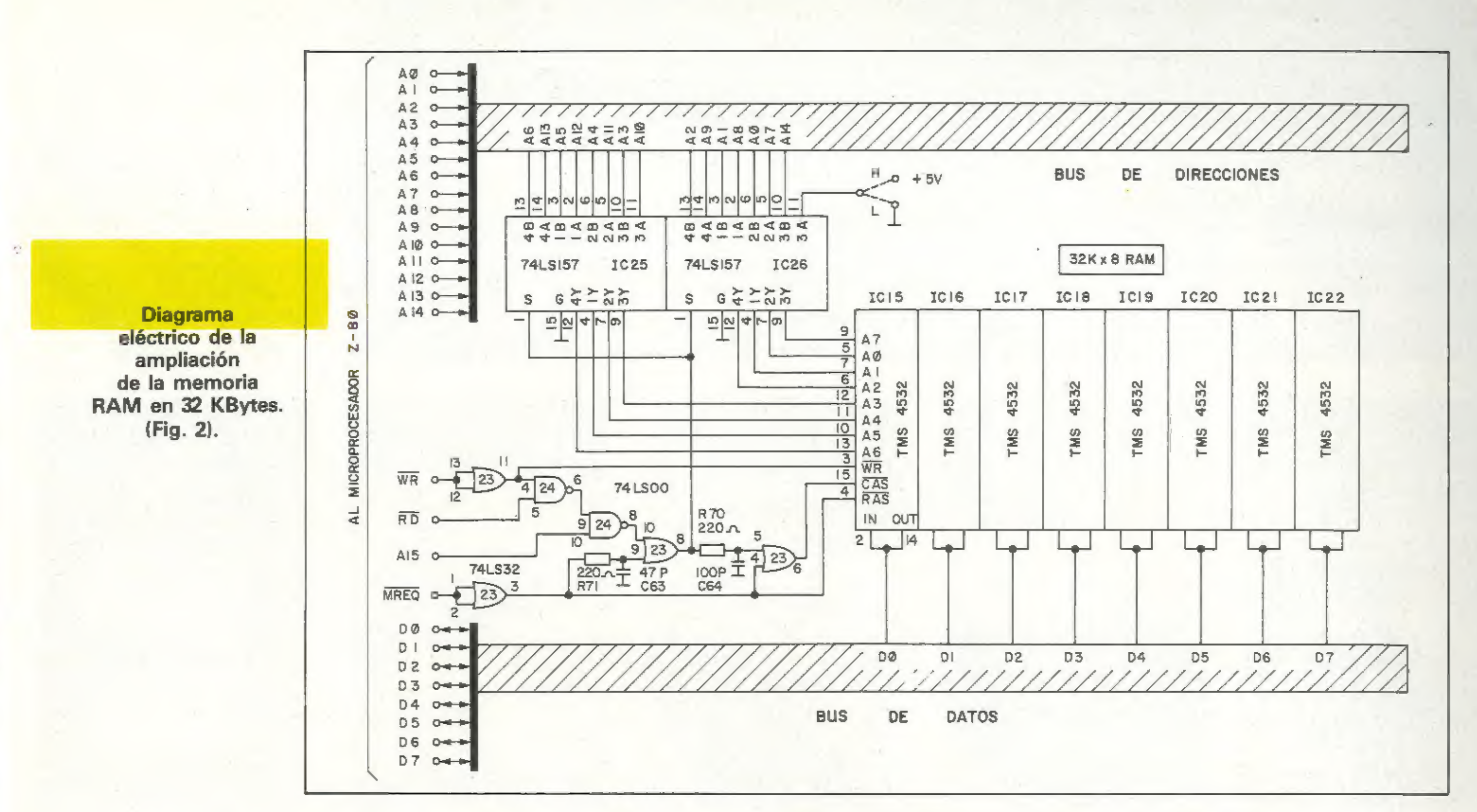


Diagrama eléctrico de la ampliación de la memoria RAM en 32 KBytes. (Fig. 2).

Las memorias RAM DINAMICAS

Las memorias RAM dinámicas se diferencian de las estáticas en varios aspectos. Necesitan menos transistores para formar una celda elemental de memoria, por tanto, caben más celdas por unidad de superficie del chip. Esto obliga a aumentar el número de terminales para un mayor número de bits de dirección, inconveniente éste, ingeniosamente resuelto como veremos. Las memorias dinámicas mantienen su información gracias a la capacidad interna que existe en cada una de las celdas en donde hay un condensador asociado a un transistor; pero este condensador sólo es capaz de mantener su carga eléctrica durante aproximadamente dos milisegundos y, dado que la presencia o no de carga determina la información binaria contenida en la celda, se hace necesario recargar este condensador antes que hayan transcurrido los dos milisegundos: este proceso se denomina refresco. En cada ciclo de refresco se produce una lectura del dato almacenado en el condensador seguido de una reescritura. De esta manera el dato queda realimentado.

En las memorias dinámicas empleadas en el Spectrum, el proceso de refresco se halla perfectamente integrado y tan sólo hacen falta tres señales con estructura de clock adecuadamente sincronizadas.

Las memorias estáticas, por contra, requieren más espacio en el chip, siendo su celda elemental un biestable que mantiene, sin más, su información con tal de no ser desconectada.

Su principal ventaja consiste en que su utilización es muy simple, aunque se requiere un mayor número de unidades para realizar un bloque de un mismo número de Kbytes. Pongamos un ejemplo: Para 16 K Bytes con RAM DINAMICA sólo hacen falta ocho chips de 16 K × 1 (Como la 4116) funcionando en paralelo. Para RAM estática sería preciso emplear 32 unidades de 1 K × 4 (Como la 2114).

Es obvio que en el Spectrum se haya recurrido al empleo de RAM dinámica para economizar espacio y costo, aprovechando las posibilidades de conexión para este tipo de memorias que están previstas en el microprocesador Z-8Ø.

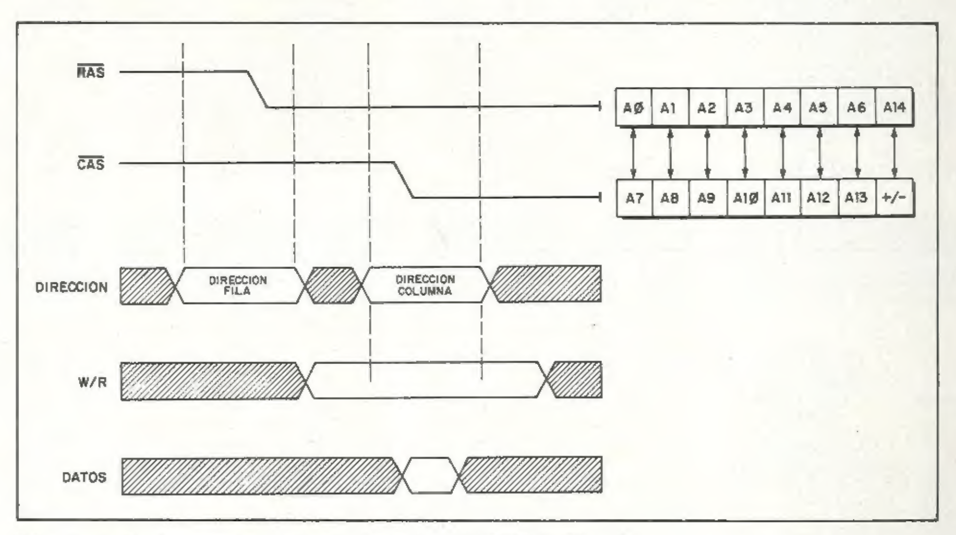
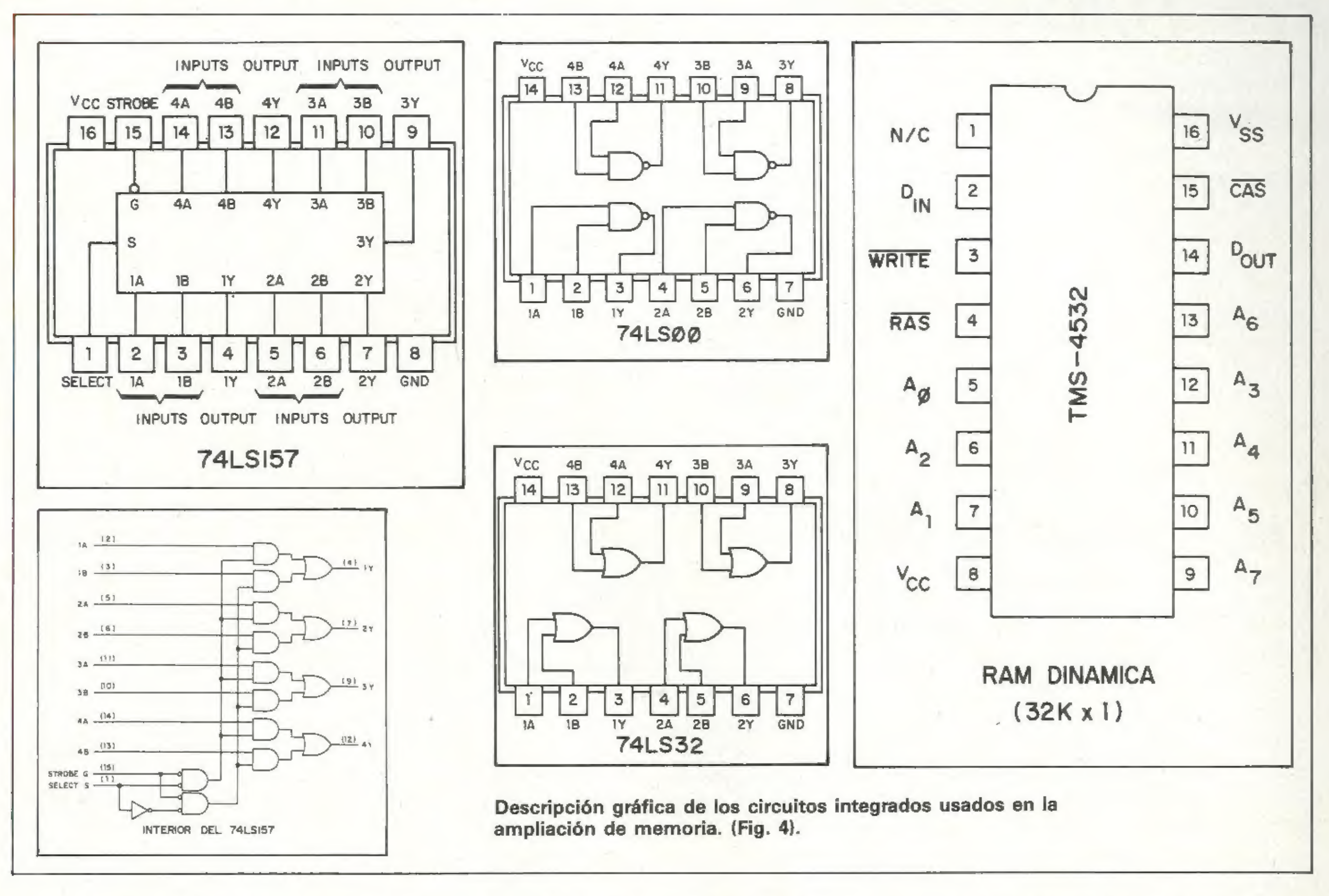


Diagrama de tiempos de la RAM dinámica TMS-4532. (Fig. 3).

Los dos tipos de memorias dinámicas comerciales utilizadas para los bloques de RAM de 16 K y 32 K, han sido las TMS 4116 de 16 K × 1 y la TMS 4532 de 32 K × 1. La primera es la más antigua de las dos y requiere de tres tensiones de alimentación diferentes: + 5V, + 12V y — 5V; la segunda tan solo necesita + 5V.

En cuanto a las señales de control, son tres, denominadas RAS, CAS y WR.



Descripción gráfica de los circuitos integrados usados en la ampliación de memoria. (Fig. 4).

RAS: Row Address Strobe (Permiso de dirección de fila).
CAS: Column Address Strober (Permiso de dirección de columna).
WR: Write/Read (Señal de escritura o lectura).

Circuito de ampliación de 32 K Bytes

Seguidamente, describiremos el circuito de ampliación de memoria. A los 16 K bytes de memoria RAM básica del Spectrum, le añadiremos 32 K bytes más, con lo que quedará convertido en un auténtico supermicroordenador de 48 K Bytes, listos, en su mayor parte, para ser utilizados por el usuario.

La memoria utilizada es la TMS 4532 de Texas Instruments, con quince bits de dirección doblados en dos ciclos (uno de ocho bits y otro de siete) para alcanzar los 32 K bytes. El doblamiento del bus de direcciones está efectuado por los dos conmutadores bidireccionales 74LS157. Cada uno de estos conmutadores es de cuatro bits, de manera que en un ciclo, introduce en los bits de direcciones de la RAM los primeros ocho bits de dirección y, en el segundo ciclo, los restantes. La entrada S de la pata 1 es la que produce la conmutación.

Llegados a este punto, le sugerimos siga el esquema de la figura dos y el diagrama de tiempos de la figura tres.

En el primer ciclo, la señal RAS introduce en la memoria los bits del bus de direcciones AØ, A1, A2, A3, A4, A5, A6 y A14 con su flanco de bajada. Este grupo de bits equivale al direccionamiento de fila, ya que los bits del interior de la memoria están organizados en forma matricial de 128 × 256.

En el segundo ciclo, la señal CAS introduce en la memoria los bits del bus de direcciones A7, A8, A9, A10, A11, A12 y A13 mediante el flanco de bajada, que ha de estar retardado respecto del de la señal RAS, unos 40 nanosegundos (40×10^{-9} SEG.).

Como se observa en la figura tres, la trasferencia del dato entre memoria y microprocesador se realiza una vez establecida la totalidad de la dirección en el segundo ciclo y en conformidad con la señal WR que determina si el ciclo es de escritura o lectura.

(Ø para escritura, 1 para lectura). Observe que en la patita número 11 del integrado conmutador IC26 existe un puente opcional que podrá ir conectado a + 5 ó a GND, según sea la variedad de la RAM adquirida. Este será el único puente a soldar por Vd. en el circuito impreso y su función es bloquear uno de los bits del segundo ciclo de doblamiento de dirección (al ser quince el número de bits (AØ8 - A14), y por tanto impares, sobra uno).

Las puertas lógicas OR y NAND de la izquierda de la figura 2 sirven para combinar las señales de control procedentes del Z-8Ø, y producir los retardos adecuados para el desfase entre RAS y CAS y evitar que se solapen señales. La doble red RC de retardo, que se encuentra en el camino de la señal CAS (R61 con C63 y R70 con C64), asume este cometido.

El bit A15 es el encargado de direccionar este bloque de RAM tal como se dijo anteriormente. La señal MREQ tiene aquí doble sentido: por una parte, sirve para dirigirse a la memoria en los ciclos de escritura o lectura y por otra, cuando no se está accediendo a este bloque de 32 K, sirve de señal de reloj que moverá constantemente los ciclos de refresco mediante la señal RAS, la cual refresca la memoria actuando sobre las columnas. Por esta razón, MREQ penetra directamente sin retardo alguno, viéndose tan sólo amplificada en potencia por una puerta OR.

CONSULTORIO

Ampliación de memoria para Spectrum 1

Tengo un Spectrum de la serie 1, aunque lo adquirí en España por medio de la firma Sitelsa, S.A.

Quisiera saber si existe o está disponible en España la ampliación de 16 a 48 K. En caso de que tenga que adquirirla en el extranjero, ruego me faciliten la dirección, precio y forma de conseguir dicha ampliación.

Si se comercializa en España, infórmenme de los mismos datos.

También quisiera saber (porque he leído algo referido a ello en alguna revista) si hay televisión pantalla, es decir, que pueda recibir la señal modulada de la televisión y también aceptar una señal de vídeo compuesto o RGB, pero que sea en blanco y negro.

Díganme el precio también, si lo saben.

Julio SANTOS GARCIA - San Sebastián

■ La ampliación de memoria para el Spectrum número 1 también es posible. La señal de vídeo que entrega el Spectrum tiene dos formatos: por un lado, sale ya modulada y, por otro, mediante el conector posterior, sale en forma de R-y, B-y e y que son los componentes de color para los monitores que se conectan a estos terminales. Para un monitor blanco y negro, puede usar la salida denominada VIDEO, que es la suma de color, luminancia y sincronismo.

Los precios de los monitores son muy dispares; pero uno bueno de color puede estar alrededor de las 80.000 ptas.

Copia analógica

Puesto que las cassettes graban en magnetofones, no sería más certero para hacer copias de seguridad conectar la cassette a copiar y una cassette virgen por medio de dos magnetofones unidos por un cable, sin usar el ordenador, como si fueran cassettes musicales.

J. A. MOZOTA - Madrid

■ El método de copia de cassettes es el conocido como copia analógica, que suele dar buenos resultados siempre que los magnetofones fuente y destino estén correctamente acoplados y sean de una mínima calidad. Este procedimiento resulta impescindible cuando se pretende copiar cintas altamente protegidas.

El sistema empleado por el programa Microcopi, utiliza la memoria del Spectrum como método digitalizador intermedio que soluciona el problema de la degrabación de señal cuando ésta se proteja únicamente de forma analógica.

Deterioro del Spectrum

Hace un año que tengo un Spectrum 48 K. y les escribo por que he notado que desde hace algún tiempo, la capa de esmalte negro que recubre el ordenador se está desconchando debido a la temperatura que llega a alcanzar tras unas horas de funcionamiento. Así, en la parte izquierda del teclado se puede observar ya una zona sin pintura que va de arriba abajo, y que, si bien de momento no afecta al teclado, temo que se extienda hasta éste y borre la nomenclatura de las teclas. ¿Qué puedo hacer para solucionarlo?: ¿puede llegar a afectar al funcionamiento del ordenador? Muchas gracias por todo.

Gabriel PIZÁ VALLESPIR
Palma de Mallorca

■ La pintura negra puede desconcharse, en efecto, pero no por la temperatura de su interior, sino por el rozamiento y la manipulación del aparato. Esto no afecta para nada su funcionamiento, al menos hasta que se desprendan también los rótulos de los comandos, ya que en ese caso sí le dificultarían su uso. La única solución por tanto es adquirir un nuevo teclado.

Norma EIA RS-232-C

Los artículos de su revista son muy interesantes, espero que continúen en esa línea.

Les sugiero que en próximos números publiquen algún artículo sobre la norma EIA RS-232-C ó V-24, o bien un resumen sobre el protocolo para acoplar un interface serie según esta norma al Spectrum.

Gracias por su amabilidad. Atentamente.

Antonio DE LA CUESTA - Cádiz

■ El interface 1 incluye una salida serie normalizada RS-232 para comunicarse con periféricos que usen este sistema.

Este mismo Interface incluye todo el Hardware y Software de acceso a cinta de microdrive. No obstante tendremos en cuenta su observación.

Transistores TR4 y TR5

He montado el Reset para mi Spectrum y funciona de maravilla; pero tengo un problema: hace algún tiempo debí introducir equivocadamente los cables de la fuente de alimentación, así que me quedé sin cables para el Spectrum ¿puedo hacer programas y ejecutarlos?

He comprobado que los trasmisores TR4 y TR5 están mal, también cambié D16 y D12, pero no sé que equivalentes hay para el TR4 ZTX 651Ø65Ø, así que tengo el Spectrum inutilizado de momento.

A ver si vosotros me lo podéis conseguir o decirme qué equivalente hay en el mercado. Gracias.

Diego ALCALA - Zaragoza

■ Resulta extraño, tras el catastrófico suceso de la inversión de polaridad de alimentación en su Spectrum, que tan sólo no le funcione la generación de color. Ha tenido suerte.

Los transistores TR4 y TR5, aunque tengan un código que pueda resultar extraño, tienen sustitutos. El TR5 ha de ser PNP y, además, capaz de soportar unos 5ØØ MA de colector (IC). Estos dos transistores tienen la misión de conformar un convertidor para obtener + 12 V, — 12 V y — 5 V a partir de 9 V.

Basic y código máquina

Soy un asiduo lector de la revista MICROHOBBY, y quiero felicitarle por ella.

Me gustaría saber en qué se parecen y en qué se diferencian el lenguaje BASIC y el CODIGO MAQUINA. También quisiera saber cómo realizar un programa con el ordenador, (me voy a comprar un Sinclair ZX81), si cada tecla tiene varias funciones distintas.

El ordenador Sinclair ZX81 ¿es el adecuado para principiantes?

Gracias.

Antonio RODRIGUEZ MARQUEZ, Córdoba

■ El lenguaje Basic y el código máquina se diferencian prácticamente en todo. Para que lo entienda mejor, el Basic es un lenguaje intérprete que ha sido creado para que nosotros podamos comunicarnos con la máquina de una forma más sencilla, mientras que el código máquina, al que se le conoce también como código objeto, es el único que entiende verdaderamente la máquina, de ahí su enorme rapidez.

Realizar un programa para ordenador es una tarea que lleva tiempo, además es necesario seguir algún curso. Nosotros le aconsejamos que siga el que publicamos semanalmente en nuestra revista.

Respecto al ZX81, es un ordenador pensado para principiantes aunque, hoy día, ya ha quedado algo anticuado y ha sido ampliamente superado por su predecesor, el ZX Spectrum.

Interface 2

1. El INTERFACE 2 tiene 32 KB de memoria. Me gustaría saber si esa memoria tiene la misma dirección de inicio que la ampliación de 32 K o si tiene una dirección diferente; es decir, si se conecta un SPECTRUM 48 K, ¿éste se queda con 48 K o aumenta su memoria a 80 K?

2. Si existen bases de datos públicas en España y si se puede acceder a ellas mediante un MODEM.

3. Los programas del n.º 1, ¿irán con la cinta de diciembre o en solitario?

Antonio BLANCO - Guadalajara

■ La dirección de memoria del Interface 2, es un celoso secreto guardado por Investrónica. De todos modos, y contestando a su pregunta, dicha dirección no puede ser la misma que para la ampliación de 32 K, ya que el Interface 2 no afecta para nada a la RAM, sino que, por el contrario, actúa directamente sobre la ROM de su Spectrum, dependiendo por tanto de la cantidad de memoria del cartucho que se utilice.

Existen bastantes bases de datos que se comercializan actualmente en nuestro país, aunque ninguna de ellas puede utilizarse con Moden.

Los programas del número 1 al 4 pertenecen al mes de noviembre, por lo que lógicamente, los del número 1 irán junto al resto de los publicados durante este mes.

¿Spectrum Plus o QL?

Pienso comprarme un ordenador y quisiera saber cuál de estos dos me aconsejan: Spectrum QL o Spectrum Plus, cuáles son, (o serán) los precios de venta al público en nuestro país, y si se pondrán a la venta antes de Navidad.

Vicente GEA MARTINEZ - Castellón

■ Aconsejarle cuál es el ordenador más de acuerdo a sus necesidades, es algo muy difícil, ya que todo dependerá del tipo de aplicaciones al que destine su uso y, por supuesto, de la cantidad de dinero que está Vd. dispuesto a gastar. Nosotros tan sólo podemos limitarnos a explicarle un poco por encima, cuáles son las ventajas de cada uno.

El Spectrum plus, tendrá un precio aproximado en torno a las 55.000 ptas. y sus características son las mismas que las del Spectrum, si bien, el teclado del plus es sensiblemente mejor. Este ordenador está indicado para aquellas personas que quieran iniciarse en el mundo de los ordenadores y en este sentido cumple perfectamente con su cometido.

El QL es un ordenador con unas prestaciones superiores a las del Plus, digamos, para que lo entienda, que es algo intermedio entre los ordenadores de gestión y los de uso doméstico. La capacidad de este último es de 128 K, frente a los 48 del Plus.

El dilema entre elegir uno y otro se lo explicaríamos quizá mejor, utilizando una metáfora. Vd. puede elegir entre un camión y un coche, el primero le va a ofrecer una capacidad mucho mayor que la del coche, pero este último le servirá perfectamente si lo que Vd. quiere es trasladarse de un sitio a otro. Imagínese que el coche es el Spectrum Plus y el camión el QL.

El precio del QL está en torno a las 130.000 ptas., aproximadamente y seguramente no estará a la venta hasta el mes de Enero. El Spectrum Plus ya está a la venta.

DE OCASION

- VENDO Spectrum 48 K. Completo. A estrenar. Cuatro publicaciones (las mejores del mercado). Doce juegos (simulador vuelo, ajedrez, The Hobby, Horacio, etc.). Precio 32.000. Sin las publicaciones 30.000. Informes. Antonio Montes Fernández C/ Comandante Vallespín, 60, 6.º C. Oviedo Tfno.: 25 93 17.
- VENDO ordenador Texas Instruments modelo 99/4 A. 16 K RAM. 32 K + 16 K (Avanced Basic en cassette) ROM. Teclado profesional. JOY STICK. Manual Avanced Basic (en inglés) cassette juego FROGGER en Basic (hecho por mí). 75.000 ptas. Tfno: 442 52 61 (Madrid). Enrique García Mauriño.
- VENDO ordenador ZX Spectrum 16 K por 26.000 ptas., comprado en Navidades y en perfecto estado, con manual en castellano. Regalo también un libro sobre Código Máquina, revistas y todo tipo de programas: Cyrus, Iet, Set, Manic Miner, Bandera a cuadros, JetPac, PSST, Simulador, Desensamblador y una larga lista. Interesados llamar al tfno.: 354 83 20. Barcelona. Preguntar por Jesús.
- Intercambio programas para el Spectrum 16/48 K. con usuarios de Baleares. Juan A. Carbonell. Plaza París, 7-4.ºB. 07010-Palma de Mallorca. Tel.: 20 14 91.
- Intercambio programas para ZX Spectrum 16/48 K: Manic Miner, Jet Set Willy, Jet Pac, Jugador de Ajedrez, La pulga, Atic Atack y un larguísimo etc. Dirigiros a: José A. Fort. Cuesta San Agustín, n.º 4-b-7, drch. La Coruña. Tel.: (981) 20 99 92. Enviar vuestra lista.
- Deseo intercambiar programas para el ZX Spectrum de 16 y 48 Kb, tengo más de 150 títulos entre los que figuran los números uno de las listas inglesas. Interesados llamar de 6 p.m. a 7 p.m. o de 9 p.m. a 10,30 p.m. O mandar una carta a Alberto Sánchez H. C/ Cánovas del Castillo, 5,1.ºJ. Alcalá de Henares. Madrid.
- Compro Interfacl 1 y pareja de joystics, todo funcionando. 10.000 y 3.000 pts. Javier Belenguer. Artes Gráficas, 16. 46010-Valencia.
- Vendo o cambio por periféricos para el Spectrum, ZX-81 con módulo de 16 K, libro de instrucciones, cables, alimentación, etc. Regalo cintas con muy buenos programas. También TV portatil B/N. Se estudiarán ofertas. Luis Amado Rego. C/ Puente, 25-3. Lalin - Pontevedra.
- Me gustaría cambiar cintas con programas del Spectrum, que sean de 16 K. para ello me podéis llamar por teléfono al número: 265 23 24, ó escribirme a Ruy González de Clavija. Madrid, 29006. Preguntar por Jesús Marquez de Clavijo. Por favor, la gente que me escriba para intercambiar cintas que viva en Madrid.

CORREO

SOBRE EL «LISTADO DE CINTAS»

El programa aparecido en la página 10 del número 2, es tremendamente útil y funciona a la perfección. Pero como he estado enredando en él, envío unas sugerencias en torno al scroll de pantalla.

1. Si en la línea 31Ø se cambia el IF contador 17 y luego PRINT AT 19,Ø pasa a PRINT AT 2Ø,Ø, se consigue una línea más en pantalla sin que aparezca el mensaje «SCROLL?»

2. EL RANDOMIZE USR 319Ø salta a la posición de ROM 3582, que es donde se encuentra en realidad la subrutina de «scroll», por lo que sería más correcto técnicamente poner RANDOMIZE USR 3582.

Esto se consigue con
LD B,18 6,18
CALL 3584 2Ø5,Ø,14
RET 2Ø
que sólo ocupa 6 bytes y congela las líneas de cabecera.

4. El PRINT AT 19,Ø es igual a un POKE 23689,4
PRINT AT 2Ø,Ø es igual a un POKE 23689,3

Resumiendo, he modificado el programa como sigue:

```
31Ø LET contador = contador + 1: IF contador 17THEN RANDOMIZE USR 3ØØ14: POKE 23689,3
345 DATA 6,18,2Ø5,Ø,14,2Ø1
35Ø FOR n = 3ØØØØ TO 3ØØ19: READ X:POKE n,X: NEXT n: RETURN
```

```
310 LET contador=contador+1: IF
 contador>17 THEN RANDOMIZE USR
30014: POKE 23689,3
 345 DATA 6,18,205,0,14,201
 350 FOR n=30000 TO 30019: READ
x: POKE n,x: NEXT n: RETURN
```

3. Precisamente esta rutina lo primero que hace es cargar en el registro B el n.º de líneas a «scrollar». Hace LD B, 23.

Si se introduce en B un n.º de líneas inferior, las líneas de arriba (TOP) de la pantalla quedan «congeladas», lo que permite evitar el PRINT AT 1, Ø; d S.

5. Cuando el contador sea 99 y se ejecute la línea 32Ø sí aparecerá el temido mensaje «Scroll?». Pero esto no me he puesto ha resolverlo. ¡¡No tengo cintas con tantos archivos!!

Zacarias Sánchez
Cartagena